The Optimisation of Hydrodynamic Vortex Separators for Removal of Solids from Wastewater, using the Continuous Adjoint Method with Topology Modification

Submitted by Shenan Grossberg to the University of Exeter as a thesis for the degree of Doctor of Engineering in April 2017.

Assuming that one has the ability to predict the performance,
the question then arises of how to modify the design to improve the performance.
(Jameson, 1988)

Abstract

Hydrodynamic vortex separators (HDVSs) are used in wastewater treatment to separate solids from wastewater. The aim of this research is to devise a CFD-based methodology that optimises their performance through modification of their design.

A validation study is performed to assess whether OpenFOAM® can be used to reliably model the flow of water in an HDVS. A one-tenth scale model is constructed and velocity measurements are recorded at 120 locations. The flow is then modelled using the Reynolds-averaged Navier-Stokes equations, and the efficacies of various mesh structures and turbulence models are investigated. The results of the simulations are compared with experimental readings, showing a good fit when the appropriate boundary layer height and turbulence model are used.

The continuous adjoint method is employed to derive the adjoint equations, associated with the drift flux equations used to model the flow of wastewater. A Darcy porosity term is included to enable a topological modification of the HDVS, and the equations are specialised to optimise the device for the removal of solids from wastewater. They are further specialised to the typical boundary conditions of ducted flows and are coded using OpenFOAM®.

An optimal design is found for boundary conditions, corresponding to typical values used in practice, and is shown to improve the performance of a simplified initial design by 40%. This optimal design is subsequently subjected to a different hydraulic loading rate and dispersed-phase volume fraction at the inlet, to assess the performance variation in these circumstances. Though the optimal design removes all the solids when the dispersed-phase fraction is reduced at the inlet, initial results suggest that the design is sensitive to hydraulic loading rate and further tests are recommended before drawing more explicit conclusions.

This is the first time the adjoint drift flux equations have been derived. It is also the first time they have been coded and applied to an HDVS to optimise its performance. The methodology developed in this thesis could be applied to any device that separates solids from liquid or two immiscible liquids, in order to optimise its performance.

Acknowledgements

I would like to express my gratitude to my supervisors, Prof. Gavin Tabor and Dr. Daniel Jarman, for their support and guidance throughout the course of this research, and to the latter for proposing this original work.

I am also indebted to Profs. Andrew Gilbert, Mark Savill, Peter Challenor and Hrvoje Jasak and to Dr. David Boger for helping me to develop the concepts used in this thesis, and to Ian Moon, Trevor Reece and George Ricketts for helping me to build a model HDVS.

I am grateful to my colleagues, Recep, Ben, Matthew, Steven, Matt, David, Pedro and Adam, for helping me to understand CFD principles and for having a good time along the way, and to Alma, David, Mark and Jacq for helping me with programming-related issues.

Finally, I would like to thank Prof. Bob Pond for his invaluable help, proofreading and suggesting edits to this thesis.

Contents

List of Figures

List of Tables

Nomenclature

Greek

α	Dispersed-phase volume fraction
β	Adjoint variable corresponding to the drift equation
Γ	Boundary
Δ	Vector between c and p
Δ_{n}	Normal distance between c and p
δ	Variation, finite increment
ϵ	Turbulent dissipation rate
η	Plastic viscosity
θ	Non-orthogonality
κ	Permeability
λ	Lagrange multipliers (adjoint variables)
μ	Dynamic viscosity
ν	Kinematic viscosity
ρ	Density
σ	Standard error
τ	Shear stress
τ_y	Yield stress
Φ	Exponential function
Φ_Γ	Function of u_{n} at the inlet and underflow
ϕ	Represents $\mathbf{u}_{\mathsf{t}}$, u_{n}, β and B

Ω Flow domain

ω Specific turbulent dissipation rate

Hebrew

$\aleph$ Porosity

Roman

A Cross-sectional area

A Operand of the diffusion stress term in the mixture momentum equation

a, b, c Constants

a, **b** Vectors

B β expressed as a mass fraction

B Operand of the term containing settling velocity in the drift equation

C Scale factor

C_1, C_2 Coefficients in the adjoint boundary conditions

C_{d} Drag coefficient

C_μ Turbulent model constant

c, c_1, c_2 Cell centres

D Diameter

D Strain rate tensor

$\mathcal{D}$ Dispersed-phase mass-flow rate

E_1, E_2 Coefficients used in coding the adjoint boundary conditions for B at the wall, underflow and overflow

e_i Vector with 0 in all entries except for 1 in the i^{th} entry

F, **F** Force

$\mathcal{F}$ Flow equations

f, f_i Frequency reading

$\overline{f}$ Mean frequency reading

f_{ns} Non-settleable fraction at the inlet

f	Common face between two cells
G	Surface-normal component of $\nabla\phi_{\mathsf{f}}$
g, **g**	Acceleration due to gravity
H	Depth
H_∞	Depth after a long period of time, i.e. order of 24 hours
h	Small perturbation
I	Initial turbulent intensity
J	Objective function
K	Turbulent diffusion coefficient
k	Settling parameter
k_1	$k \ln 10$
k	Turbulent kinetic energy
L	Characteristic length
$\mathscr{L}$	Lagrange function
$\mathcal{L}$	Hydraulic loading rate
l	Turbulent length scale
M	Matrix
M	Number of design parameters
N	Number of flow variables
n	Number of readings
n	Vector normal to the face
n	Component of vector normal to the face
ñ	Vector normal to the edge
$\mathcal{O}$	Order of magnitude
P	Particle
$\mathcal{P}$	Performance
P	Pressure

p Kinematic pressure

p_{rgh} Total pressure minus hydrostatic pressure

p, p_1 Face centres

p_2 Intersection of the face and the line c_1c_2

Q Flow rate

q Adjoint pressure

$\mathcal{R}$ Residuals of the primal (flow) equations

$R_1, \cdots, R_5$ Residuals of the steady state drift flux equations including Darcy term

Re Reynolds number

r_f Flocculent settling parameter

r_h Hindered settling parameter

S_1, S_2 Source terms in the adjoint drift equation

sg Particle specific gravity

t Time

t_0 Time zero

u, **u** Adjoint velocity

V Volume

v, **v** (Primal) velocity

v_0, $\mathbf{v}_0$ Maximum theoretical settling velocity

v_{00} Maximum practical settling velocity

v_{dj}, $\mathbf{v}_{dj}$ Dispersed-phase settling velocity in layer j

v_{dm} Dispersed-phase settling velocity relative to the mixture velocity

v_p Particle settling velocity

w Weighting

X Concentration of suspended solids

X_{min} Concentration of non-settleable solids at the inlet

x, y, z Cartesian x, y, z directions

x Design parameters

y Flow variables

y^+ Normalised distance to the wall

Subscripts

$\aleph$ Pertaining to the porosity

α Pertaining to the dispersed-phase volume fraction

Γ Pertaining to the boundary

Ω Pertaining to the flow domain

b Pertaining to body

c Pertaining to the continuous phase

c Pertaining to the cell

$crit$ Pertaining to a critical value

cyl Pertaining to the cylinder

d Pertaining to the dispersed-phase

d Pertaining to drag

E Pertaining to particles at the end of a test

F Pertaining to the full-size geometry

$\overline{f}$ Pertaining to the mean frequency reading

f Pertaining to the face

G Pertaining to the gradient-evaluated part

g Pertaining to gravity

i, k Counter variables

in Pertaining to the inlet

j Pertaining to layer j

M Pertaining to the model geometry

m Pertaining to the mixture

m Pertaining to capillary

n	Pertaining to the normal component
o	Pertaining to the overflow
p	Pertaining to the particle
p_m	Pertaining to the mixture pressure
T	Pertaining to particles released into the system
t	Pertaining to time
t_0	Pertaining to time zero
t	Pertaining to the tangential component
u	Pertaining to the underflow
V	Pertaining to the fixed value part
$\mathbf{v}_m$	Pertaining to the mixture velocity
x, y, z	Pertaining to the Cartesian x, y, z directions
x	Pertaining to the design parameters
y	Pertaining to the flow variables

Superscripts

$'$	Fluctuating component of the Reynolds decomposition
-1	Inverse
$\overline{}$	Time-averaged
T	Transpose
t	Turbulent

Abbreviations

APWA	American Public Works Association
CFD	Computational fluid dynamics
CPU	Central Processing Unit
CSO	Combined sewer overflow
EPA	Environment Protection Agency
FOG	Fats, oil and grease
HDVS	Hydrodynamic vortex separator
LRR	Launder-Reece-Rodi
NSE	Nash-Sutcliffe efficiency
PBIAS	Percent bias
RHS	Right hand side
RST	Reynolds stress transport
SST	Shear stress transport
WWTP	Wastewater treatment plant

Chapter 1

Introduction

The water cycle is a continuous process by which water circulates between the Earth's oceans, atmosphere and land, and is a vital process for sustaining life on this planet. Humans interfere with this cycle to obtain water for their homes, businesses and industries. In the developed world, water that has been used is transported to a wastewater treatment plant, where the contaminants are removed and the wastewater is made environmentally safe, before being returned to the receiving watercourse or sea.

Wastewater treatment typically involves four stages: preliminary, primary, secondary and tertiary (Metcalf and Eddy, 2003). Preliminary treatment removes all the materials that can be easily collected, such as tree branches, grit or plastic bags, before they clog the system or damage the pumps. Primary treatment uses quiescent settling basins to remove organic settleable solids, through sedimentation, while substances which can float are skimmed off the surface. Secondary treatment uses aerobic bacterial action to remove the remaining organic matter, and requires aeration which is the most energy-intensive operation in wastewater treatment. Tertiary treatments consist of various forms, including filtration and disinfection, to further improve the effluent quality before it is discharged to the receiving environment.

Sedimentation is commonly performed in large rectangular tanks but can be performed more efficiently, using lamella plates or hydrodynamic vortex separators (HDVSs) (Dudley, 1994). The main issue with rectangular tanks is that they take up a large amount of space, while lamella plates can clog due to shock loadings of fats, oil and grease. HDVSs avoid both these issues. The development of HDVSs dates back to the 1960s, when Bernard Smisson, an engineer from Bristol, UK, developed the first full-size cylindrical vortex chamber to treat the combined sewer overflow (CSO) in Bristol (Smisson, 1967). At the time, the regulations regarding effluent discharge were not as stringent as they are now. Nevertheless, Smisson was so concerned that untreated effluent could be discharged into the environment that he used his living room to develop his early HDVS prototypes.

In the 1970s, based on Smisson's initial work and with his help as a consultant, the American Public Works Association (APWA) and US Environment Protection Agency developed a second generation HDVS to remove settleable solids from CSO, for which Smisson received 'an APWA award of merit in recognition of the excellence of his work, ingenuity, resourcefulness, and grasp of hydraulic engineering principles' (Andoh and Smisson, 1993). By the early 1990s, HDVSs were also used in municipal wastewater treatment, industrial effluent treatment and stormwater treatment, receiving favourable reviews in Europe, North America and Japan (Andoh et al., 2002). By 2004, over 1,500 had been installed worldwide, ranging in size from 0.75 m to 16 m in diameter (Faram et al., 2004) though it is important to emphasise that the design of each HDVS is for a particular application.

Before the advent of computational fluid dynamics (CFD), the only way to improve the performance of a device was through empirical trial and error. However, this was time-consuming and expensive, and there was no certainty that a modified design would be optimal. CFD is an interdisciplinary topic that uses fluid mechanics, numerical analysis and computer science to solve and analyse problems that involve fluid flow. Based on the Navier-Stokes equations and improvements in computing performance, CFD has been used since the 1950s to model the flow of fluids with increasing accuracy and complexity (Harlow, 2004). Coupled with the improvements in computing performance, developments in computer code have enabled the progression from two-dimensional to three-dimensional methods (Hess and Smith, 1967), from subsonic to transonic speeds (Jameson and Caughey, 1977) and from single-phase to multiphase flows (Lyczkowski, 2010). Simulations could be repeated relatively easily on different designs, and algorithms were developed to search the design space, ranging from global optimisation algorithms, such as genetic algorithms, to gradient-based algorithms, such as finite differencing. As CFD simulations are relatively time-consuming to solve, the algorithms used are generally gradient-based, with the most prominent being the adjoint method because it is independent of the design space dimension. Therefore, it is worth emphasising that an optimal design found using the adjoint method may be a local rather than a global optimum.

1.1 Aims and Objectives

Though CFD has been used to optimise the performance of aeroplanes, trains, ships and automobiles, it has not been applied in the water industry, partly because the industry is not considered sufficiently hi-tech to warrant hi-tech approaches (Faram and Harwood, 2000) and partly because the equations that govern the flow of wastewater are not the same as are used in the sectors where the

optimisation techniques have been applied. The adjoint Navier-Stokes equations have been derived and applied to modify the shape of objects, through which the flow is governed by the Navier-Stokes equations (Othmer, 2008). However, the flow of wastewater is most appropriately described by the drift flux equations (Brennan, 2001; Burt, 2010). In order to apply the adjoint method to a device though which wastewater flows, the adjoint drift flux equations would need to be derived and applied in a way that has not been seen before. It would also be necessary to develop a methodology that uses the results of the adjoint drift flux equations to modify the shape of the device to improve its performance.

The aim of this research was to derive the adjoint drift flux equations and to devise a CFD-based methodology that optimises the performance of an HDVS through modification of its design. This was achieved through realising the following objectives:

1. Investigation of the background information and previous work (Chapter 2) required to inform the theory development and methodology including:
   (a) A review of the sedimentation processes used in wastewater treatment, focussing on the use of CFD with HDVSs to determine whether this research has been previously attempted.
   (b) Evaluation of the mathematical models used to describe the flow of wastewater, most notably the drift flux model.
   (c) Assessment of the physical characteristics of wastewater relevant to the drift flux model.
   (d) Understanding shape modification techniques, concentrating on topology modification because it can be used to change the internal shape of a device.
   (e) Analysis of optimisation techniques, in particular the continuous adjoint method because it requires fewer iterations to reach an optimal solution.
2. A validation study was performed to assess whether OpenFOAM® could be used to reliably model the flow of water through an HDVS (Chapter 3). A one-tenth scale model was constructed and set it up in a laboratory to receive the same inlet velocity a full-size HDVS would receive. Velocity measurements were recorded at 120 locations and repeated five times to obtain average velocity readings. The flow was modelled using the Reynolds-averaged Navier-Stokes equations, and the efficacies of various mesh structures and turbulence models were investigated. The results of the simulations were compared with the experimental readings, and the findings from this study were used to inform the simulations performed in the remainder of this research.

3. The continuous adjoint drift flux equations, including a Darcy porosity term, were derived so that the internal shape of a device could be modified to optimise its performance (Chapter 4). The equations were further specialised to the typical boundary conditions of an HDVS and the outcomes tested with an objective function, being the removal of solids from the flow of wastewater. The topological sensitivities of the objective function at each cell were computed, to determine whether they were favourable or counterproductive for the flow, and a procedure was proposed for determining which cells to remove from the domain. The adjoint drift flux system was then discretised and coded using OpenFOAM®, employing the Robin condition at boundaries formed from a combination of Dirichlet and Neumann boundary conditions.
4. A proof of concept was carried out (Chapter 5) to demonstrate the effectiveness of the methodology developed in Chapter 4. The methodology was applied to a simplified HDVS, to find an optimal HDVS design for boundary conditions corresponding to typical values used in practice. This design was then subjected to a different hydraulic loading rate and dispersed-phase volume fraction at the inlet, to assess the performance variation in these circumstances.
5. CFD was used to explain the effect of the optimal design on the flow of wastewater, resulting in the improved performance; the methodology was also critically reviewed and measures for its improvement were indicated (Chapter 6).
6. The implications of this research were considered and recommendations for further work were proposed (Chapter 7).

All the laboratory work, theory development, computer coding and simulation runs in this thesis were performed by the author.

Chapter 2

Background and Review of Previous Work

The objectives of this chapter are to describe the background and previous work relevant to the present study and to explain the selection of mathematical protocols adopted in Chapter 4, so that the performance of hydrodynamic vortex separators can be optimised. It has been divided into five sections: sedimentation processes in wastewater treatment, physical characteristics of wastewater, modelling the flow of wastewater, shape modification techniques and shape optimisation techniques for sedimentation devices.

2.1 Sedimentation Processes

2.1.1 Wastewater Treatment Plants

Sedimentation in wastewater treatment plants (WWTP) can be carried out in a number of ways. The most common method is to use a settling tank, Figure 2.1. However, such tanks can be costly to build, take up a large amount of potentially valuable space and the process can be very slow. Another method is to use lamella plates, Figure 2.2, where wastewater flows up through the plates and the solids settle on their surfaces, slide down to the base of the tank, while the effluent rises to the top of the tank and flows over the weir. However, their performance can be compromised by shock loading and by fats, oil and grease (FOG) build-up between the plates.

An alternative is to use a hydrodynamic vortex separator (HDVS), Figure 2.3, in which wastewater enters the tank tangentially at around mid-depth. Tangential entry and the internal geometry induce rotational flow, Figure 2.4, causing solids to flow in a downward helical path in the outer region of the tank, while the effluent rises in an upward helical path in the central region to the top and flows out of the tank, Figure 2.5. The HDVS is cheaper to build and uses less space than a

settling tank, because of its greater depth and higher hydraulic loading rate, and has a higher resistance to shock loading and FOG build-up than lamella plates, because there are fewer small spaces for FOG to accumulate.

Figure 2.1: Settling tank (City of Akron, 2016).

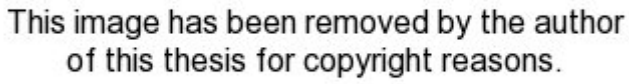


Figure 2.2: Lamella plates (CST Wastewater Solutions, 2014).

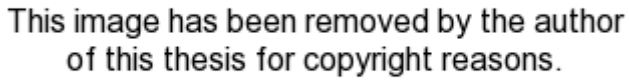


Figure 2.3: Hydrodynamic vortex separator, diameter 5.2 m,
(Faram et al., 2004, p.86).

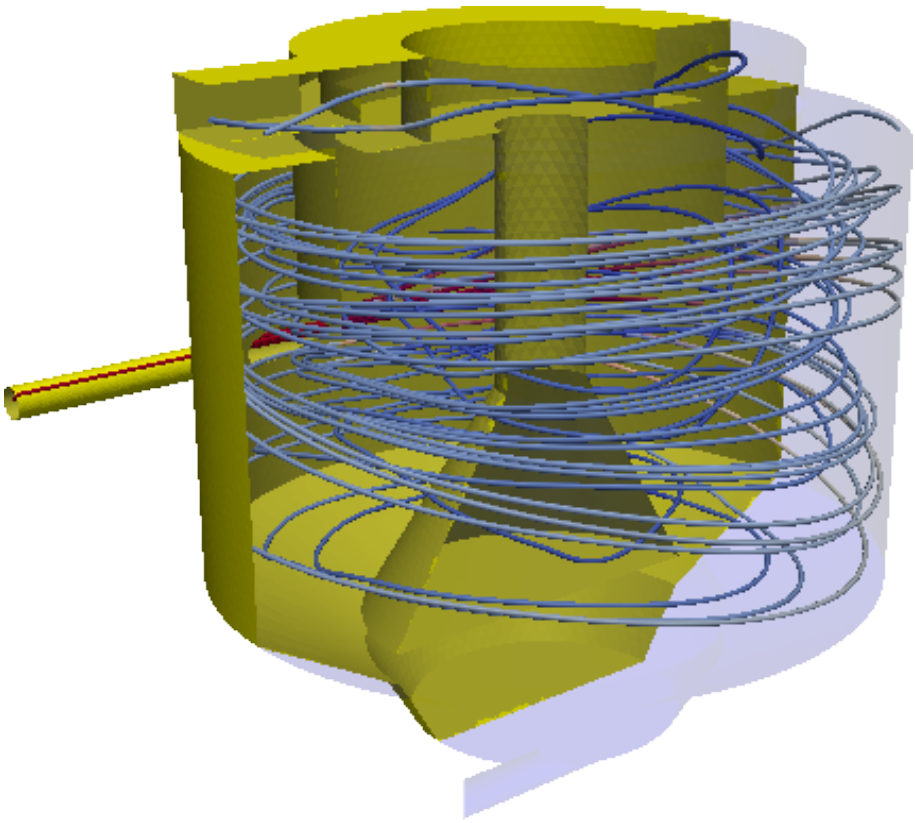


Figure 2.4: Cutaway drawing of an HDVS, revealing internal geometry and flow pathlines.

Figure 2.5: Simplified flow pattern in an HDVS, showing outer and inner helical spirals of the solids and effluent, respectively, (Andoh et al., 2002, p.5).

WRc, independent consultants working in the water sector, compared the performance of the Swirl-Flo^{TM} Separator, a variant of the HDVS, with lamella plates at constant flow rates (Dudley, 1994). They concluded that the Swirl-Flo had a comparable performance to lamella plates and settling tanks, while being operated at twice the hydraulic loading rate. They also asserted that the Swirl-Flo should perform better than the other devices in wastewater treatment works, where the flow rate varies due to the adverse effects of shock loading.

2.1.2 Combined Sewer Overflows

The majority of sewer networks transport wastewater and stormwater to WWTP, with overflows constructed along the network to prevent overloading the system during wet weather events. Traditionally, it was considered acceptable to discharge diluted wastewater from these combined sewer overflows (CSOs) into receiving watercourses without treatment. This is no longer the case and discharges in the UK must now be treated according to the European Union Urban Waste Water Treatment Directive.

The problem with many CSOs is that they are usually located in confined spaces, where it is difficult to incorporate treatment processes. The HDVS is an ideal solution, because of its small footprint, high hydraulic loading rate and robust design. An example of an HDVS being used in CSO treatment can be seen in Figure 2.6, where raw sewage is transported to the diversion structure and excess flows are diverted into the new CSO treatment facility, containing the Storm King® HDVS. The clarified overflow is discharged into the Saco River, while the underflow, comprising the captured settleable solids, is returned to the plant for treatment. A new grit removal facility, containing the Grit King® HDVS, has been added to the plant to remove grit before the wastewater is treated in the downstream processes.

Figure 2.6: Aerial view and flow scheme at Saco WWTP
(Andoh et al., 2011, p.3).

2.1.3 Use of CFD with HDVSs

Prior to 2000, computational fluid dynamics (CFD) had often been regarded as 'a hi-tech tool for the solution of hi-tech problems' (Faram and Harwood, 2000). Faram and Andoh (2000) provided examples of where CFD had been and could be used effectively in the water industry, focussing on where CFD had been used to assess the efficiency of an HDVS intended for primary sedimentation applications. Comparison of data with the predictions of a well validated semi-empirical model suggested that appropriate CFD software is effective for assessing the impacts of changes to an HDVS. Faram and Harwood (2002, 2003) presented a methodology for the assessment of stormwater sediment interceptors to retain and remove sediments using CFD and concluded that primitive chambers, such as gully pots, are less likely to retain sediments during high flow inputs than advanced systems, such as HDVSs. Jarman et al. (2007) reviewed the usage history of CFD techniques by a supplier of technologies for urban water management, and concluded that CFD methods offer genuine opportunities and could yield direct tangible benefits. Jarman et al. (2008) further presented a review of CFD studies carried out in the analysis of urban drainage systems, including CSOs, storage and attenuation systems, stormwater sediment interceptors and sewerage conveyance structures.

Egarr et al. (2004, 2005b,c) used CFD to model the flow in an HDVS in order to predict the residence time of the fluid at the outlets. The results were compared with those determined experimentally and demonstrated that it is possible to use CFD to predict the mean residence time of the fluid and that it is possible to apply these techniques to predict the mean survival rate of bacteria in a combined separation and disinfection process. Egarr (2005) performed a sensitivity study of HDVSs using CFD and discovered that particle surface load, not settling velocity, controls their efficiency. Residence time distribution (RTD) models were assessed and it was found that the most suitable model is the axial dispersion model. Validation of residence time characteristics was reasonable, which allowed CFD to be used to describe the disinfection performance of an HDVS. With the scaling of residence time to larger separators having been developed, it was concluded that CFD had given an insight into the fluid dynamics within the HDVS. Egarr et al. (2005a) and O'Doherty et al. (2009a) furthered this work and demonstrated that the theoretical performance of an HDVS as a contact vessel for disinfection can be predicted and confirmed the practical applicability of an HDVS for disinfection. O'Doherty et al. (2009b) compared four RTD models with CFD modelling to determine which is most suitable for representing the RTD of an HDVS when compared to RTD measurements taken under laboratory conditions.

Li (2009) developed a sizing equation to improve the performance of HDVSs. Experimental investigation determined that inlet pipe elevation had insignificant impact on particle removal efficiency, whereas chamber height had a significant

and measurable effect. A simple formula for angular velocity was derived, which was coupled with the Navier-Stokes equations to develop a vortex flow pattern model. A new particle settling formula was proposed, which was coupled with the vortex flow pattern model to derive particle trajectory equations. The sizing equation was developed from the particle trajectory equations and was validated by laboratory measured particle removal efficiencies of HDVSs. It was concluded that the sizing equation would improve the performance of HDVSs, however no other modifications to the HDVS were explored.

Pathapati and Sansalone (2009) used CFD to model multiphase flow in HDVSs, employing a k-ϵ turbulence model and Lagrangian tracking for the discrete phase. They validated their results against pilot-scale data and demonstrated less than 10% difference between model and measured data. Sansalone and Pathapati (2009) then compared these CFD results against event-based flow statistics, which are commonly used to analyse the performance of HDVSs, and showed that the CFD model yielded an accurate predictive capability for particulate matter separation compared with event-based flow statistics. Ying et al. (2012) concluded that 'CFD is arguably the most powerful tool available' for the modelling of stormwater treatment.

Schmitt et al. (2014) investigated the use of CFD to predict the solids' separation efficiency of an HDVS, employing an Eulerian-Lagrangian approach to model the trajectory of the particles. They focussed on the particles' trajectories near the screen and compared two types of screen, highlighting the turbulent effects created by the shape of the screen. Gronowska-Szneler (2015) investigated the velocity field inside a vortex separator and developed formally simple and technically justified design criteria that could be conveniently used in the design process.

CFD was used in all of these research cases to improve the understanding of flow within HDVSs and, in some cases, to improve the performance of HDVSs through sizing the chamber. However, none of the cases attempted to modify the shape of HDVSs to improve their performance, or to optimise their performance. The objectives of this research are to use CFD to: *a*) derive equations to modify the shape of HDVSs that improve their performance, and *b*) develop a methodology that optimises their performance. The remaining sections in this chapter prepare the background and review previous work in order to develop the theory and methodology in Chapter 4.

2.2 Physical Characteristics of Wastewater

2.2.1 Settleable Solids

Solids in wastewater are classified as dissolved or suspended, the latter generally having a diameter greater than 0.45 μm (UK Technical Advisory Group, 2012). Suspended solids are further classified as settleable, if they settle out of suspension within a limited period of time, usually one hour. Typically about 60% of suspended solids in municipal wastewater are settleable (Metcalf and Eddy, 2003) and have been classified into five types, Table 2.1, reproduced from (Crabtree, 1989, p.570). A typical sequence of sediment deposits in a combined sewer pipe can be seen in Figure 2.7. Though the classification applies to sewers, Types A, C and D are mobile and would be transported to a WWTP, where the denser, inorganic solids are usually removed before the organic solids.

Table 2.1: Settleable solids in a combined sewer pipe.

Type	Description
A	Coarse, loose, granular, predominantly mineral, material found in the inverts of pipes.
B	As A, but concreted by the addition of fat, bitumen, cement, etc. into a solid mass.
C	Mobile, fine grained deposits found in slack flow zones, either in isolation or above Type A material.
D	Organic pipe wall slimes and zoogloeal biofilms around the mean flow level.
E	Fine-grained mineral and organic deposits found in storm sewage overflow (SSO) storage tanks.

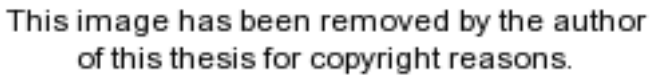


Figure 2.7: A typical sequence of sediment deposits in a combined sewer pipe (Crabtree, 1989, p.574).

2.2.2 Settling Velocity

Gravity settling of organic solids has been categorised into four stages, depending on the concentration and tendency of the particles to interact (Metcalf and Eddy, 2003) as summarised in Table 2.2.

Table 2.2: Settling stages of organic solids.

Settling Stage	Concentration	Description
Discrete particle	Low	Unhindered settling where particles settle as individual entities.
Flocculent	Low-intermediate	Particles coalesce (flocculate) increasing in mass and settling at faster rate.
Hindered	Intermediate	Inter-particle forces hinder settling of neighbouring particles, reducing the settling rate.
Compression	High	Particles form a structure and only settle through compression of the structure.

Discrete Particle Stage

The information in this section is taken from Metcalf and Eddy (2003). The terminal velocity of a discrete particle, P, represented in Figure 2.8, can be determined by equating the gravitational force of the particle, F_g, with the drag force acting on it,

F_d, as shown in Eqns. (2.2.1),

$$F_\mathrm{g} = F_\mathrm{d} \tag{2.2.1a}$$

$$\implies (\rho_\mathrm{p} - \rho_c) V_\mathrm{p} g = \frac{1}{2} \rho_c A_\mathrm{p} C_\mathrm{d} {v_\mathrm{p}}^2 \tag{2.2.1b}$$

$$\implies v_\mathrm{p} = \sqrt{\frac{4 D_\mathrm{p} g (sg - 1)}{3 C_\mathrm{d}}}, \tag{2.2.1c}$$

where:

- ρ_c is the continuum density,
- ρ_p is the particle density,
- $sg = \dfrac{\rho_\mathrm{p}}{\rho_c}$ is the particle specific gravity,
- V_p is the particle volume,
- A_p is the particle cross-sectional area,
- D_p is the particle diameter, assuming a spherical particle,
- C_d is the drag coefficient,
- v_p is the particle settling (drift) velocity and
- g is the acceleration due to gravity.

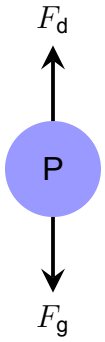


Figure 2.8: Forces acting on a discrete particle, P.

The drag coefficient takes on different values, depending on the flow regime, and is approximated by,

$$C_\mathrm{d} = \frac{24}{Re_\mathrm{p}} + \frac{3}{\sqrt{Re_\mathrm{p}}} + 0.34, \tag{2.2.2}$$

where $Re_\mathrm{p} = \dfrac{v_\mathrm{p} D_\mathrm{p}}{\nu_c}$ is the particle Reynolds number, $Re_\mathrm{p} < 10^4$, and ν_c is the continuum kinematic viscosity. For transitional flow, the complete form of Eqn. (2.2.2) must be used to determine the settling velocity. For viscous flow, $Re_\mathrm{p} < 1$,

the first term in Eqn. (2.2.2) predominates and substituting it into Eqn. (2.2.1c) yields the settling velocity for Stokes flow,

$$v_{\mathsf{p}} = \frac{(sg-1){D_{\mathsf{p}}}^2 g}{18\nu_c}. \tag{2.2.3}$$

For turbulent flow, the effect of the first two terms is reduced and can be approximated by substituting $C_{\mathsf{d}} = 0.4$ into Eqn. (2.2.1c) which yields,

$$v_{\mathsf{p}} = \sqrt{3.33 D_{\mathsf{p}} g(sg-1)}. \tag{2.2.4}$$

Discrete Particle, Flocculent and Hindered Stages

The settling velocity in the other stages is harder to predict and, in the flocculent and hindered stages, the usual procedure is to carry out a settling column test. Nevertheless, Vesilind (1968) derived Eqn. (2.2.5) to model the dispersed-phase settling velocity in hindered settling conditions,

$$v_{dj} = v_0 e^{-k\alpha}, \tag{2.2.5}$$

where:

- v_{dj} is the dispersed-phase settling velocity in layer j,
- v_0 is the maximum theoretical settling velocity,
- k is a settling parameter and
- α is the dispersed-phase volume fraction.

Dahl (1993) achieved a closer fit to the published settling velocity and dispersed-phase concentration data by using Eqn. (2.2.6),

$$v_{dj} = v_0 10^{-k\alpha}, \tag{2.2.6}$$

where the coefficient and exponent can be seen in Table 2.4. Takacs et al. (1991) adapted Eqn. (2.2.5) creating Eqns. (2.2.7) to model the settling velocity of activated sludge across different layers,

$$v_{dj} = v_0 e^{-r_h(X_j - X_{min})} - v_0 e^{-r_f(X_j - X_{min})}, \tag{2.2.7a}$$

$$0 \leqslant v_{dj} \leqslant v_{00}, \tag{2.2.7b}$$

where:

- r_h is the hindered settling parameter (m^3/g),
- r_f is the flocculent settling parameter (m^3/g),

- X_j is the concentration of suspended solids in layer j (g/m^3),
- X_{in} is the concentration of suspended solids at the inlet (g/m^3),
- f_{ns} is the non-settleable fraction of suspended solids at the inlet,
- $X_{min} = f_{ns}X_{in}$ is the concentration of non-settleable solids at the inlet and
- v_{00} is the maximum practical settling velocity.

The first term in Eqn. (2.2.7a) reflects the settling velocity of the large, well-flocculating particles, while the second term is a velocity correction factor to account for the smaller, slowly settling particles. The parameters, used in Eqns. (2.2.7), were modified by Coderre (1999) so that they could be applied to primary settling and are reproduced from Gernaey et al. (2001, p.75) in Table 2.3.

Table 2.3: Parameters used in Eqn. (2.2.7) for primary settling.

Settling parameter	Value	Settling parameter	Value
v_0	96 m/day	r_h	0.00019 m^3/g
v_{00}	80 m/day	r_f	0.0007 m^3/g
f_{ns}	0.24		

A plot of settling velocity versus concentration of suspended solids is reproduced from Takacs et al. (1991, p.1266), Figure 2.9, where settling zones II, III and IV cover the discrete particle, flocculent and hindered settling stages.

Compression Stage

The depth of the compression region at time, t, can be determined from Eqn. (2.2.8),

$$H_t - H_\infty = (H_{t_0} - H_\infty)e^{-c(t-t_0)}, \tag{2.2.8}$$

where H_t is the depth at time, t, H_{t_0} is the depth at time, t_0, H_∞ is the depth after a long settling period, i.e. the order of 24 hours, and c is a constant for the given suspension (Metcalf and Eddy, 2003).

2.2.3 Viscosity

The continuum viscosity, μ_c, is the constant of proportionality relating shear stress, τ, and shear rate, $\frac{\partial v_x}{\partial y}$,

$$\tau = \mu_c \frac{\partial v_x}{\partial y}, \tag{2.2.9}$$

where v_x is the component of velocity in the x direction (Symon, 1971). The dispersed-phase is said to exhibit plastic viscosity, η, which is defined as the con-

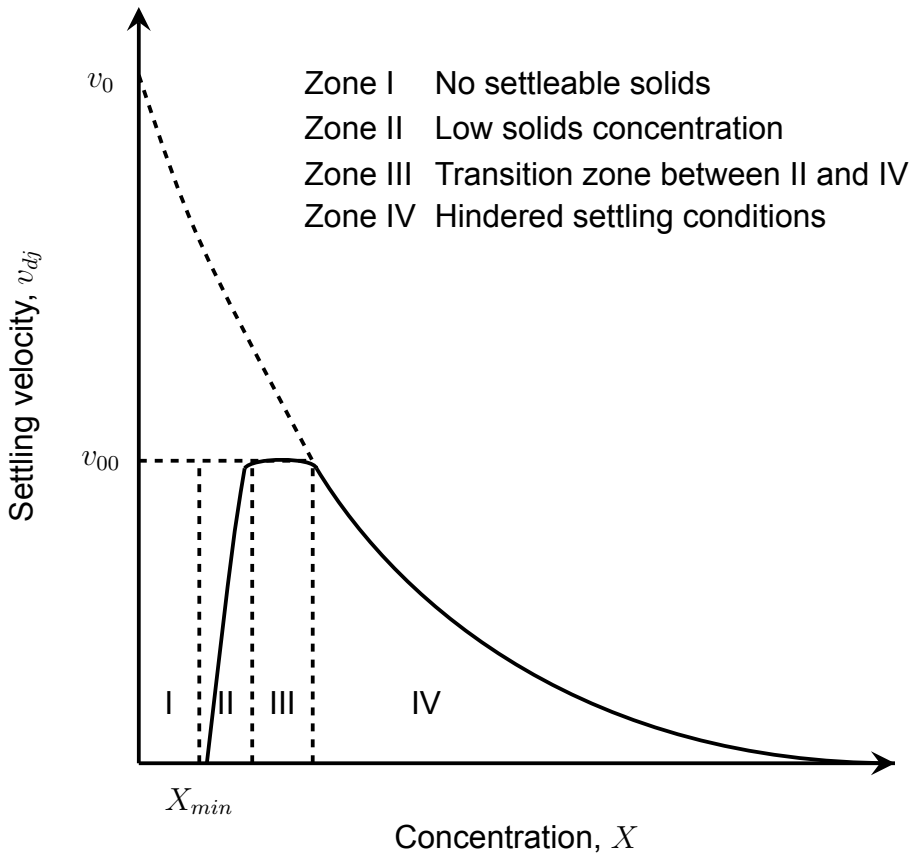


Figure 2.9: Plot of settling velocity versus concentration of suspended solids.

stant of proportionality between $\tau - \tau_y$ and shear rate,

$$\tau - \tau_y = \eta \frac{\partial v_x}{\partial y}, \tag{2.2.10}$$

where τ_y is the yield stress and $\tau \geqslant \tau_y$ for the mixture to be in motion (Bingham, 1922). The constants, μ_c, η and τ_y, are graphically represented in Figure 2.10.

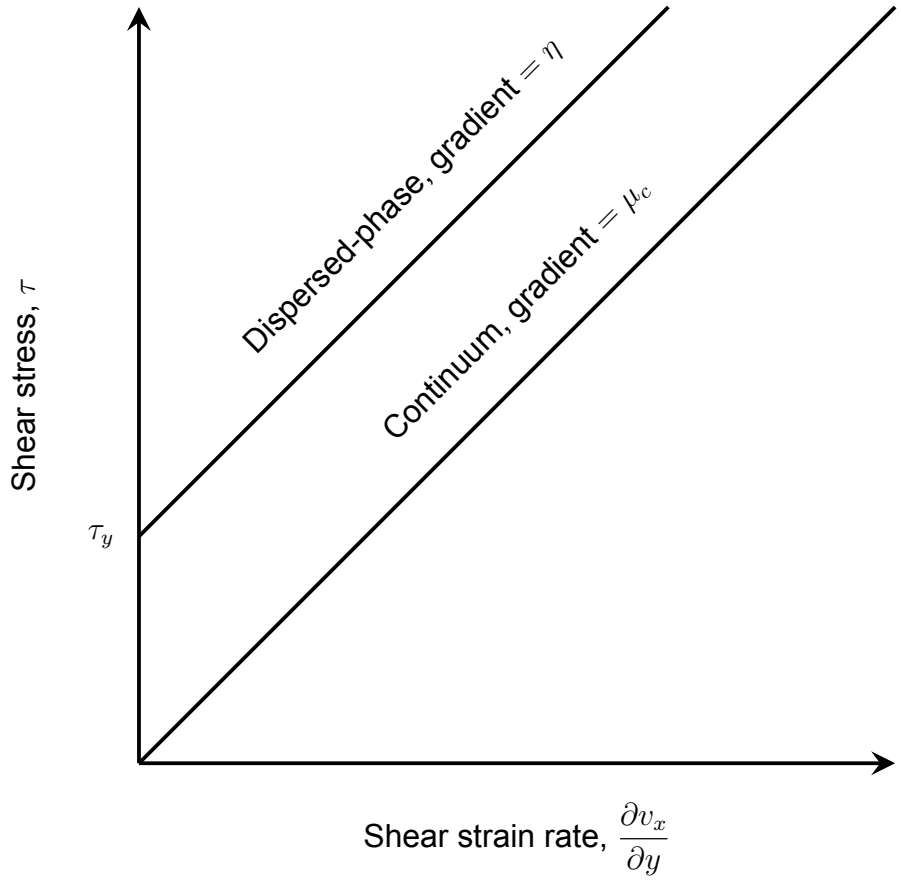


Figure 2.10: Graphical representation of constants, μ_c, η and τ_y.

Therefore, in order to calculate the dispersed-phase viscosity, η and τ_y need to be determined. They are modelled using the general exponential form (Brennan, 2001),

$$\Phi = a10^{b\alpha}, \tag{2.2.11}$$

where Φ represents η or τ_y, and a and b are functions of the dispersed-phase volume fraction at the inlet, α_{in}, the dispersed-phase density, ρ_d, and the initial turbulent intensity, I, which is expressed as a percentage of the inlet velocity. This is the same form of equation, used in Section 2.2.2, to model settling velocity. Comparing Eqns. (2.2.6) and (2.2.11), we find $v_0 = a$ and $k = -b$. The values of a and b used in Dahl (1993) were based on ρ_d = 2,000 kg/m^3 and I = 10%. They were shown to yield good predictions of the velocity and α distributions and are reproduced in Table 2.4.

Table 2.4: Values of a and b used in Dahl's experiment.

	a	b ($\alpha_{in} = 0.1\%$)	b ($\alpha_{in} = 0.2\%$)
Yield stress (kg/m/s^2)	5.5469×10^{-5}	1050.80	951.25
Plastic viscosity (kg/m/s)	2.3143×10^{-4}	179.26	179.26
Settling velocity (m/s)	-2.198×10^{-3}	-285.84	-285.84

2.3 Modelling the Flow of Wastewater

Wastewater consists of two phases: a solid (dispersed) phase and a liquid (continuum) phase, which are assumed to be isothermal, incompressible and without phase change. There are three ways to model the flow of wastewater:

1. Eulerian-Eulerian model, also known as the two-fluids model,
2. Eulerian-Lagrangian model, also known as particle tracking, and
3. Drift flux model, also known as the mixture model.

They are summarised as follows.

2.3.1 Eulerian-Eulerian Model

In the Eulerian-Eulerian model, each phase is modelled in full and, therefore, the momentum and continuity equations are solved for each phase. They are coupled through terms in the momentum equations, which describe the interfacial forces and transfer momentum between them.

The advantage of this model is that a fully deterministic approach to settling can be adopted by calculating, for example, the drag force across a full range of particle sizes, using the correlations of Clift et al. (1978) and Ishii and Zuber (1979). The particle interactions in the flocculent stage can be modelled by increasing the effective mixture viscosity (Gidaspow, 1994) and the drag correlations in the hindered and compression zones can be modelled by applying semi-empirical formulations (Ergun, 1952; Witt and Perry, 1995).

However, the solid phase in this model is defined in terms of a single length scale, shape factor and density, which is not congruent with the solid phase in wastewater. For this reason, and the difficulty of describing the formation and deformation of larger agglomerations in the hindered and compression zones, the Eulerian-Eulerian model is not used in this research.

2.3.2 Eulerian-Lagrangian Model

In the Eulerian-Lagrangian model, the liquid phase is solved in the Eulerian frame and the solid phase is solved in the Lagrangian frame. The difference between the two reference frames is that the Eulerian frame focusses on specific locations in space, while the Lagrangian frame focusses on particles passing through space.

As with the Eulerian-Eulerian model, coupling exists between the phases and can be modelled either as one-way or two-way coupling. In one-way coupling, the momentum of the liquid phase impacts on the solid phase, but not vice versa, and the particles simply follow the flow of the liquid phase. In two-way coupling,

the momentum of each phase impacts on the other, through drag, lift, buoyancy or other forces.

The problem with this model is that, when particles interact with a wall, they terminate and are removed from the system, whereas physically they remain and interact with the body of flow (Burt et al., 2002). As the dispersed-phase volume in an HDVS is an essential component of this research, the Eulerian-Lagrangian model has been rejected.

2.3.3 Drift Flux Model

The information in this section is taken from Brennan (2001). In wastewater, the density ratio between the two phases is low, generally less than 2:1, and the drag force between them is high. Therefore, to a good approximation, the two phases can be considered to respond to pressure gradients as a single phase. Additionally, the slip (drift) between the phases is primarily due to the gravitational settling of the dispersed phase.

In the drift flux model, the two phases are treated as one: the momentum and continuity equations for both phases are summed to create a mixture-momentum and mixture-continuity equation, and the transport of the dispersed phase is modelled using a drift equation. The latter is derived from the dispersed-phase continuity equation, which uses the dispersed-phase diffusion velocity, defined as the dispersed-phase velocity relative to the mixture velocity, v_{dm}, but the drift equation uses the dispersed-phase drift velocity, defined as the dispersed-phase velocity relative to the mixture centre of volume, v_{dj}, because it can be measured in batch settling tests. They are defined relative to each other as,

$$v_{dj} = \frac{\rho}{\rho_c} v_{dm}, \tag{2.3.1}$$

but, as the flow field in these tests is laminar, a turbulent diffusion term is included in the drift equation and modelled according to the Boussinesq hypothesis. The three equations, collectively called the drift flux equations, are now listed below,

$$\frac{\partial \rho_m \mathbf{v}_m}{\partial t} + (\mathbf{v}_m \cdot \nabla)(\rho_m \mathbf{v}_m) = -\nabla P_m + \nabla \cdot \left(2\mu_m \mathsf{D}(\mathbf{v}_m)\right) - \nabla \cdot \left(\frac{\alpha}{1-\alpha}\frac{\rho_c \rho_d}{\rho_m}\mathbf{v}_{dj}\mathbf{v}_{dj}\right) + \rho_m \mathbf{g} + \mathsf{F_m}, \tag{2.3.2a}$$

$$\frac{\partial \rho_m}{\partial t} + \nabla \cdot (\rho_m \mathbf{v}_m) = 0, \tag{2.3.2b}$$

$$\frac{\partial \alpha}{\partial t} + \nabla \cdot (\alpha \mathbf{v}_m) = -\nabla \cdot \left(\frac{\alpha \rho_c}{\rho_m}\mathbf{v}_{dj}\right) + \nabla \cdot K \nabla \alpha, \tag{2.3.2c}$$

where the terms, not previously defined, are:

- μ_m, the mixture viscosity, defined as the sum of the continuum, dispersed-

phase and turbulent viscosities,

- ρ_m, the mixture density, defined as $\alpha\rho_d + (1-\alpha)\rho_c$,
- $\mathbf{v}_m$, the mixture velocity,
- $\mathrm{D}(\mathbf{v}_m) = \frac{1}{2}\left(\nabla\mathbf{v}_m + (\nabla\mathbf{v}_m)^T\right)$, the mixture strain rate tensor,
- K, the turbulent diffusion coefficient, defined as the mixture eddy diffusivity, ${\nu_m}^t = \frac{{\mu_m}^t}{\rho}$, where ${\mu_m}^t$, the mixture turbulent viscosity, is obtained from turbulence modelling,
- $\mathbf{F}_\mathrm{m}$, the capillary force and
- $P_m = \rho_m p_m$, the mixture pressure, where p_m is the mixture kinematic pressure.

In summing the momentum equations, not only have the number of equations been reduced from four to three, but the inter-phase momentum transfer terms have also been eliminated which were numerically unstable. Hence, a far more robust equation set has been produced and the computational resources required to solve the system have been reduced.

On the other hand, some information has been lost. The shape and deformation of the sludge flocs is not modelled and, in the vertical direction, the motion between the two phases is replaced with an experimentally derived settling velocity of the dispersed-phase. Nevertheless, the model is appropriate for the lack of detailed information regarding sludge rheology and sludge flocs and, consequently, Brennan (2001) and Burt (2010) proposed modelling wastewater with the drift flux model.

2.4 Shape Modification Techniques

The shape of a sedimentation device can be modified in two ways, either through its geometry or its topology. With the former, the existing boundary of the fluid domain is modified, whereas with the latter, new boundaries can be introduced into the design (Guest and Prévost, 2006). In both cases, modification is achieved through parametrisation of the design space and is explained, with examples, in the following sections.

2.4.1 Geometry Modification

Geometry modification involves a change in structure, defining the fluid region, and can be achieved through parametrising the geometry. An example of this is shown in Figure 2.11, which represents the cross-section of an aeroplane wing. The line defining the outline of the wing has control points (squares) on it, which can be moved to modify the geometry.

Figure 2.11: Wing outline with control points to modify geometry
(Zingg et al., 2008, p.111).

Another example is shown in Figure 2.12, where the black line depicts the cross-section of a ship hull, the red lines illustrate the flow across it and the arrows indicate the direction to modify its geometry.

Figure 2.12: Arrows indicating where to modify hull geometry
(Stück and Rung, 2011, p.7).

An example involving car design is shown in Figure 2.13, where the direction of favourable surface displacement for reducing drag is indicated by colour (red: push, blue: pull).

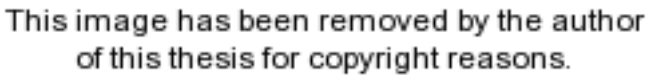
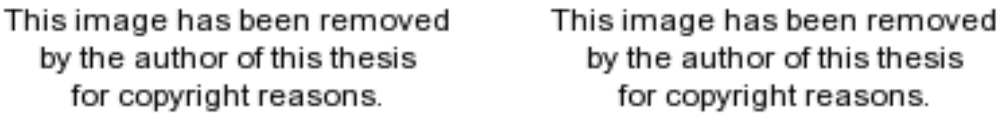


Figure 2.13: Colour map indicating direction of surface displacement (Othmer, 2014, p.12).

Geometry modification was applied to the rear of the car, Figure 2.14a, resulting in the formation of a spoiler, Figure 2.14b.



(a) (b)

Figure 2.14: Geometry modification applied to rear of (a), baseline design, resulting in formation of spoiler on (b), optimised design, (Giannakoglou, 2012, p.19).

2.4.2 Topology Modification

In solid mechanics, topology is defined as 'the pattern of connectivity [...] of elements in a structure' (Rozvany, 1997). It was introduced into fluid mechanics by Borrvall and Petersson (2003) so that a change of connectedness could be applied to a fluid region by means of an indicator. The equations in this section are taken from Guest and Prévost (2006). By combining the Stokes equations governing creeping flow,

$$\mu\nabla^2\mathbf{v} = \nabla p - \rho\mathbf{F}_\mathrm{b}, \tag{2.4.1a}$$

$$\nabla\cdot\mathbf{v} = 0, \tag{2.4.1b}$$

where $\mathbf{F}_b$ denotes the body force, with the Darcy equations which describe viscous flow through a porous medium,

$$-\frac{\mu}{\kappa}\mathbf{v} = \nabla p - \rho\mathbf{F}_b, \tag{2.4.2a}$$

$$\nabla \cdot \mathbf{v} = 0, \tag{2.4.2b}$$

where κ is the permeability, the Stokes-Darcy equations were developed,

$$\mu\nabla^2\mathbf{v} - \aleph\mathbf{v} = \nabla p - \rho\mathbf{F}_b, \tag{2.4.3a}$$

$$\nabla \cdot \mathbf{v} = 0, \tag{2.4.3b}$$

where $\aleph$ is the inverse permeability, referred to by Othmer (2008) as porosity. When $\aleph = 0$, the equations describe Stokes flow and, when $\mu = 0$, the equations describe Darcy's law with porosity, $\aleph$. The Darcy term, $\aleph\mathbf{v}$, dampens the velocity and, therefore, porosity acts as a design parameter, because it is very small in fluid regions and very large in non-fluid regions.

An example of topology modification is shown in Figure 2.15, which represents the cross-section of an air duct. The shape was being optimised for flow uniformity at the outlet (RHS) and the hot and cold colours correspond to favourable and adverse cells, respectively.

Figure 2.15: Colour map inside air duct, indicating favourable and adverse cells (Othmer et al., 2006).

Another example is shown in Figure 2.16, where the application is a flow splitter manifold, consisting of a single inlet at the base of the unit, four outlets leaving the domain at a right angle to the inlet, and a boxed section representing the possible design envelope. In this case, the shape of the manifold was being modified to minimise power dissipation and the porosity distribution identifies the areas to apply blockage (blue: none, red: maximum).

Figure 2.16: Porosity distribution inside manifold to minimise power dissipation (Othmer et al., 2007).

2.4.3 Topology or Geometry

From the literature, it would appear that geometry modification is used predominantly for flow around objects, such as aeroplanes, ships and cars, and that topology modification is used for flow through objects, such as ducts and manifolds, the reason being that topology modification provides the opportunity not only to alter the external shape, as with geometry modification, but also to introduce new boundaries into the design. The main disadvantage with topology modification is that the resultant shape usually has a ragged surface, however the technique is very efficient at creating draft designs that can be fine-tuned using geometry modification (Othmer, 2008).

As this research is concerned with flow through a sedimentation device, topology modification would appear to be the more suitable option. However, though this technique was developed for Stokes flow and then applied to the Navier-Stokes equations, it has not previously been applied to multiphase flow. Nevertheless, multiphase flow through a sedimentation device is closer to Stokes flow in terms of its velocity than flow through an air duct, so the author has decided to proceed with topology modification.

2.5 Shape Optimisation Techniques

With the ability to model the flow of wastewater and modify the design of a sedimentation device, the objective now is to modify the design to improve its performance. Options to do this are through a global or local (gradient-based) search of the design space, and are explored in this section.

2.5.1 Gradient-based versus Global Search

CFD calculations are notoriously time consuming to solve (Stück, 2012) and, as they need to be solved for each design, it is imperative to identify a methodology that finds the optimal design in as few iterations as possible. Gradient-based methods utilise the gradient of the function at the current point to determine the direction of the search, (Polak, 1997). An example of this can be seen in Figure 4.1a, where the red line represents the path of steepest ascent along the solution space from the empty-red circle at the lower end, representing an initial solution, to the filled-red circle at the upper end, representing an optimal solution. The limitation of a gradient-based search is that the optimal solution is only a local optimum, because the search is confined to the solution space surrounding the current point. Genetic algorithms offer a global search and could be used to find an innovative design solution, however they require orders of magnitude more iterations than gradient-based methods (Holst and Pulliam, 2003; Funke et al., 2014) and for this reason are not explored further in this study.

Much of the subsequent information in this section is taken from Farrell (2016). The performance of a sedimentation device can be measured through an objective function, J, and the design can be modified through parametrisation of the design space, x. It is usually relatively straight-forward to calculate the objective function and, more importantly, the sensitivity of the objective function with respect to changes in the design space, $\frac{dJ}{d\mathsf{x}}$. This derivative gives an indication of the effect the design has on the performance of a device, and gradient-based methods make use of this information. One such method is finite differencing, which uses a first-order approximation to the gradient,

$$\frac{dJ}{d\mathsf{x}} \approx \frac{J(\mathsf{x} + he_i) - J(\mathsf{x})}{h}, \tag{2.5.1}$$

where e_i is the vector with 0 in all entries except for 1 in the i^{th} entry and h is a small perturbation applied to the i^{th} component of x. The main issue with this method is that the objective function, and hence the flow equations, need to be solved for each i. The number of iterations required to obtain an optimal solution has order of magnitude, $\mathcal{O}(M)$, where M is the number of design parameters. As M is generally large, this method can be computationally expensive.

Alternatively, the tangent linear method applies the chain rule to the gradient of the objective function,

$$\frac{dJ}{d\mathsf{x}} = \frac{\partial J}{\partial \mathsf{y}} \cdot \frac{d\mathsf{y}}{d\mathsf{x}} + \frac{\partial J}{\partial \mathsf{x}}, \tag{2.5.2}$$

where y are the flow variables, and the total derivative to the flow equations, $\mathcal{F}(\mathsf{x}, \mathsf{y}) = 0$,

$$\frac{\partial \mathcal{F}}{\partial \mathsf{y}} \cdot \frac{d\mathsf{y}}{d\mathsf{x}} + \frac{\partial \mathcal{F}}{\partial \mathsf{x}} = 0. \tag{2.5.3}$$

Using Eqn. (2.5.3) to calculate $\frac{d\mathsf{y}}{d\mathsf{x}}$ and substituting it into Eqn. (2.5.2), $\frac{dJ}{d\mathsf{x}}$ is calculated without the need to solve the flow equations. However, $\frac{d\mathsf{y}}{d\mathsf{x}}$ is a massively dense matrix, of dimensions the product of the solution space and the parameter space, and is unlikely to fit in memory.

2.5.2 Adjoint Method

The adjoint method, which originated from control theory (Lions, 1971), was pioneered in the fields of fluid mechanics and aerodynamics by Pironneau (1974) and Jameson (1988), respectively, because it is independent of the number of design parameters. It is a gradient-based method that can be presented in terms of the discrete or continuous approach (Giles and Pierce, 2000).

Discrete Approach

Much of the information in this section is taken from Farrell (2016). For the discrete approach, Eqn. (2.5.3) is rearranged in terms of $\frac{d\mathsf{y}}{d\mathsf{x}}$, assuming $\frac{\partial \mathcal{F}}{\partial \mathsf{y}}$ is invertible, and substituted into Eqn. (2.5.2) as follows,

$$\frac{dJ}{d\mathsf{x}} = -\frac{\partial J}{\partial \mathsf{y}} \cdot \left(\frac{\partial \mathcal{F}}{\partial \mathsf{y}}\right)^{-1} \frac{\partial \mathcal{F}}{\partial \mathsf{x}} + \frac{\partial J}{\partial \mathsf{x}}, \tag{2.5.4}$$

where $\left(\frac{\partial \mathcal{F}}{\partial \mathsf{y}}\right)^{-1}$ denotes the inverse of matrix, $\frac{\partial \mathcal{F}}{\partial \mathsf{y}}$. Applying the following property of matrices to a matrix, M,

$$\mathsf{M}^T\mathbf{a} \cdot \mathbf{b} = \mathbf{a} \cdot \mathsf{M}\mathbf{b}, \tag{2.5.5}$$

where M^T is the transpose of M and **a** and **b** are vectors, Eqn. (2.5.4) becomes,

$$\frac{dJ}{d\mathsf{x}} = -\left(\frac{\partial \mathcal{F}}{\partial \mathsf{y}}\right)^{-T} \frac{\partial J}{\partial \mathsf{y}} \cdot \frac{\partial \mathcal{F}}{\partial \mathsf{x}} + \frac{\partial J}{\partial \mathsf{x}}. \tag{2.5.6}$$

Taking the left hand side of the dot product to be λ^T and rearranging we have,

$$\frac{\partial J}{\partial \mathsf{y}} + \lambda \frac{\partial \mathcal{F}}{\partial \mathsf{y}} = 0, \tag{2.5.7}$$

which is known as the adjoint equation.

Continuous Approach

Much of the information in this section is taken from Othmer (2008). In the continuous approach, the optimisation problem is formulated as,

$$\mathscr{L} = J + \lambda \mathcal{F}, \tag{2.5.8}$$

where $\mathscr{L}$ is the Lagrange function and λ are the Lagrange multipliers. The Lagrange function is optimised by deriving its variation with respect to the design parameters and the flow variables, i.e.

$$\delta \mathscr{L} = \delta_{\mathsf{x}} \mathscr{L} + \delta_{\mathsf{y}} \mathscr{L} \, . \tag{2.5.9}$$

Since the design parameters and the flow variables are mutually dependent, computing the variation directly from Eqn. (2.5.9) would require solving the flow equations once for each design parameter, which would be computationally very expensive. However, if the Lagrange multipliers are chosen so that the variation with respect to the flow variables vanishes, i.e.

$$\delta_{\mathsf{y}} \mathscr{L} = 0, \tag{2.5.10}$$

then the variation can be computed simply from

$$\delta \mathscr{L} = \delta_{\mathsf{x}} \mathscr{L} \, . \tag{2.5.11}$$

The adjoint equations, for a given set of flow equations, are derived from Eqn. (2.5.10) and the Lagrange multipliers, also referred to as the adjoint variables, are the solution to those equations.

In both approaches, the adjoint method has order of magnitude, $\mathcal{O}(N)$, where N is the number of flow variables, so where $N \ll M$, the adjoint method is the preferred optimisation technique. The approaches differ in their formulation; in the discrete approach, the flow equations are discretised and then linearised, whereas in the continuous approach, the adjoint equations are derived, linearised and then discretised. The main disadvantage of the continuous approach is that derivation of the adjoint equations is complex, however by deriving them, their physical significance is clearer, their solution is simpler and requires less memory. For these reasons, the continuous approach has been used in this research.

2.6 Summary

Mathematical methods for modelling the flow of wastewater in HDVSs, and optimising their design, are reviewed above. The particular protocols adopted in the present work are as follows. As set out by Brennan (2001), viscosity and the settling velocity of solids are modelled using an exponential form, Eqn. (2.2.11). The drift flux model, proposed by Brennan (2001) and Burt (2010), is employed to model wastewater flow, Eqns. (2.3.2), and is adopted as the primal equations for the adjoint derivation in Chapter 4. The shape of HDVSs is modified using the topological approach, and is optimised using the continuous adjoint method (Othmer, 2008). These protocols are utilised in Chapter 4 to derive the adjoint drift flux equations and devise a methodology to modify the shape of HDVSs, so that their performance can be optimised.

Chapter 3

Validation Study

The aims of this chapter are to assess whether OpenFOAM® can be used to accurately model the flow of water in a Hydrodynamic Vortex Separator (HDVS), and in doing so, elucidate the effects of the mesh structure and turbulence model used, and thereby inform the simulations performed in the remainder of this research. The experiment was conducted by the author, with assistance from Beatrix Haigh[1] at the Hydro International[2] laboratory on the 12th–14th February 2014. Parts of this chapter have been reproduced from Grossberg et al. (2015).

3.1 Experimental Setup

3.1.1 Scale Model

A one-tenth scale model HDVS was used in this study, Figure 3.1, having dimensions: height = 0.664 m, diameter = 0.6 m, inlet pipe length = 1 m and inlet pipe diameter = 0.028 m. The internal components of the HDVS can be seen in Figures 3.2 and 3.3, where the former shows the deflector plate and dip plate, and the latter is an exploded view, showing the dip plate, baffle plate and cone. In this experiment water flowed into the HDVS, but in practice, wastewater flows into the HDVS through the inlet pipe, entering the outer (blue) section tangentially to the outer wall. Flow is directed towards the circumference of the HDVS by the deflector plate and, as it spirals around the inside of the HDVS, solid particles fall to the bottom, while cleaner water rises to the top of the cylinder through the gap between the dip and baffle plates and out through the overflow.

[1] Year in Industry student at Hydro International

[2] Shearwater House, Clevedon Hall Estate, Victoria Road, Clevedon BS21 7RD

Figure 3.1: One-tenth scale model HDVS connected to a water source.

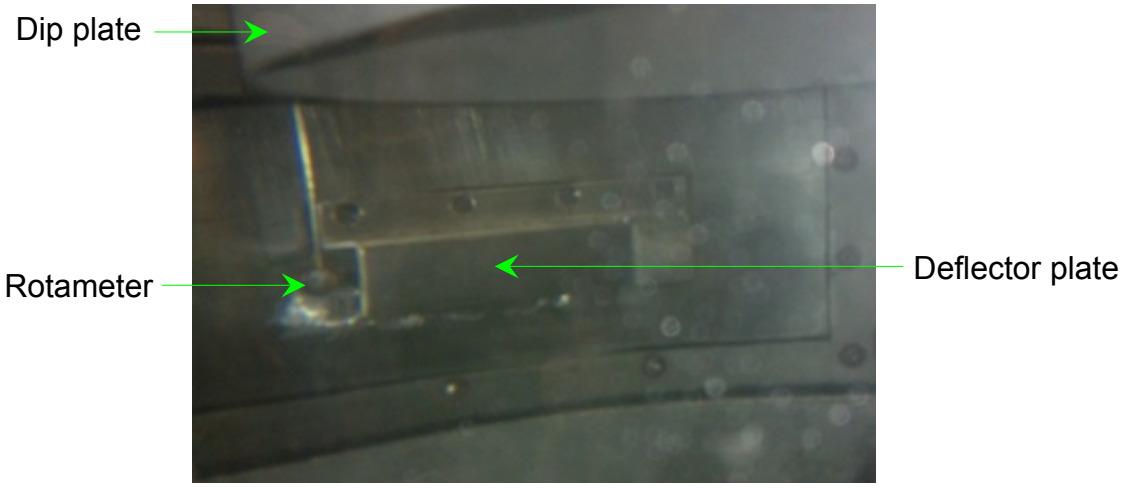


Figure 3.2: Rotameter positioned at end of deflector plate.

3.1.2 Hydraulic Loading Rate

The Reynolds number, which is the ratio of inertial forces to viscous forces in a fluid, is often used in scaling fluid dynamics problems, to determine dynamic similarity between a model and its full-size geometry. However hydraulic loading rate, not dynamic similarity, is the most important factor when scaling an HDVS, because separation of solids is mainly due to gravity rather than centripetal forces (Veerapen et al., 2005) and is relatively independent of inertial forces. The hy-

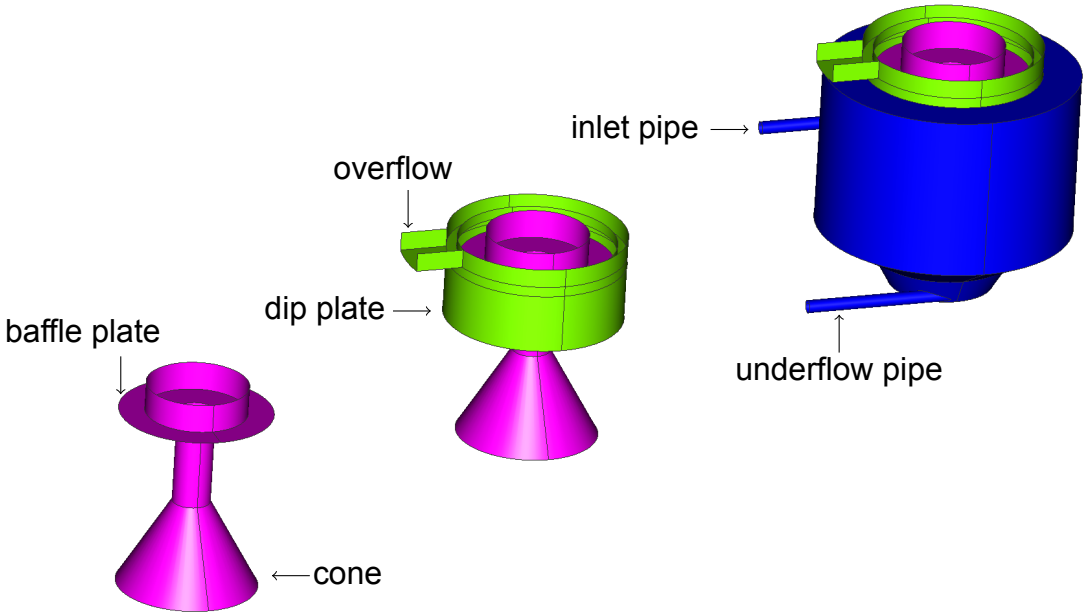


Figure 3.3: Exploded view of the HDVS.

draulic loading rate is defined as the flow rate per settling area of the device,

$$\mathcal{L} = \frac{Q}{A_{cyl}} \times 10^{-3}, \tag{3.1.1}$$

where $\mathcal{L}$ is the hydraulic loading rate (l/s/m^2), Q is the flow rate (m^3/s) and A_{cyl} is the settling area (m^2) (WEF, 2005). The hydraulic loading rate of the full-size geometry is therefore,

$$\begin{aligned} \mathcal{L}_F &= \frac{Q_F}{A_{cylF}} \times 10^{-3} \\ &= \frac{C^3 Q_M}{C^2 A_{cylM}} \times 10^{-3} \\ &= C\,\mathcal{L}_M, \end{aligned} \tag{3.1.2}$$

where F and M refer to the full-size and model geometries, respectively, and C is the scale factor between them. Similarly, the inlet velocity is defined as,

$$v_{in} = \frac{Q}{A_{in}}, \tag{3.1.3}$$

where v_{in} is the inlet velocity (m/s) and A_{in} is the inlet cross-sectional area (m^2) and, therefore, the inlet velocity of the full-size geometry is,

$$\begin{aligned} v_{inF} &= \frac{Q_F}{A_{inF}} \\ &= \frac{C^3 Q_M}{C^2 A_{inM}} \\ &= C v_{inM}. \end{aligned} \tag{3.1.4}$$

From Eqns. (3.1.2) and (3.1.4), the hydraulic loading rate and inlet velocity scale with the geometry. The only time this does not occur is when the inlet velocity would fall below the minimum velocity required to carry grit into the cylinder (EPA, 2003). Though this is not a multiphase experiment, the aim of this study is to inform the simulations of multiphase flow later in this research, and as the scaled inlet velocity would fall below the threshold, the hydraulic loading rate and inlet velocities were not scaled with the geometry. Instead, the model HDVS was connected to a water source, delivering a constant hydraulic loading rate of 3 l/s/m^2, which is equivalent to the hydraulic loading rate a full-size HDVS would receive.

The inlet velocity was calculated from the hydraulic loading rate by rearranging Eqn. (3.1.1) in terms of Q and substituting into Eqn. (3.1.3),

$$\begin{aligned} \implies v_{in} &= \mathcal{L}\, \frac{A_{cyl}}{A_{in}} \times 10^{-3} \\ &= \mathcal{L} \left(\frac{D_{cyl}}{D_{in}} \right)^2 \times 10^{-3}, \end{aligned} \tag{3.1.5}$$

where D_{cyl} and D_{in} are the cylinder and inlet pipe diameters, respectively. Therefore, the inlet velocity was 1.38 m/s.

3.1.3 Location of Velocity Readings

Velocity readings were taken at 120 locations, which consisted of ten heights, three radial distances and four azimuthal angles from the centre line of the HDVS, as shown in Figures 3.4 and 3.5. The heights were 0.03 m apart, from -0.09 m below to 0.18 m above the centre line of the inlet pipe, the radial distances were 0.01 m, 0.03 m and 0.07 m from the outer vertical wall, and the azimuthal angles subtended were 22.5°, 90°, 180° and 270° from the vertical centre line of the HDVS to the end of the inlet pipe[3].

[3] 22.5° was used to coincide with the end of the deflector plate (Figure 3.2)

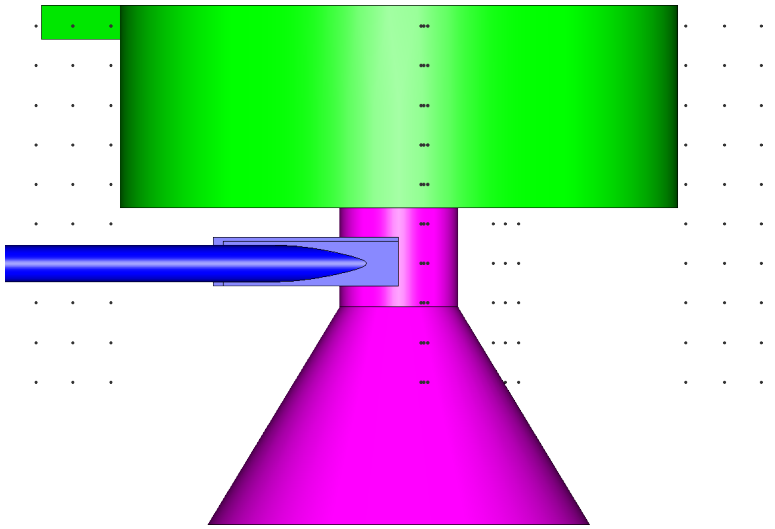


Figure 3.4: Elevation view of locations (dots) used to take velocity readings.

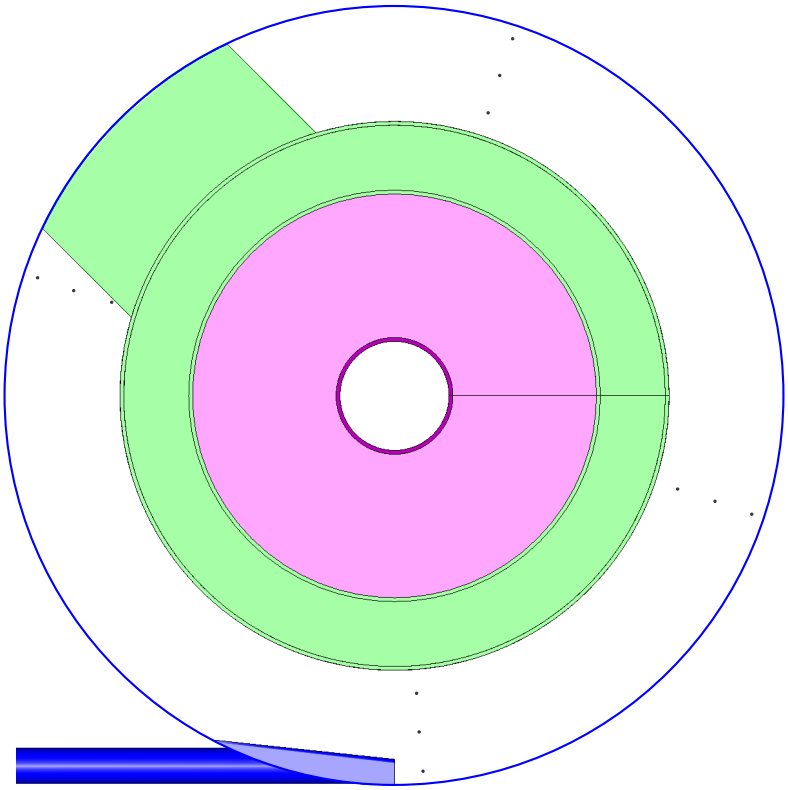


Figure 3.5: Plan view of locations (dots) used to take velocity readings.

3.1.4 Rotameter and Frequency Readings

The velocity readings were taken using a rotameter[4], Figure 3.2, connected to a control unit, which recorded an average frequency reading every ten seconds. The rotameter was positioned, oriented and held in place, using two 3D-printed jigs, Figure 3.6, while five readings were taken at each location. A calibration chart, Figure A.1, was provided to convert the rotation frequencies of the impeller

[4]Nixon Flow Meters' Streamflo Low Speed Velocity Probe 403

into velocities, however the black and red lines were not parallel, so it was ambiguous which line to choose in the range covered by both lines. A cubic spline interpolation was calculated between the two straight lines, to provide a unique solution in this range, Figure A.2, and the equations are recorded in Appendix A.1. The frequency readings are recorded in Appendix A.2.

(a) (b)

Figure 3.6: The lower jig in (a) was used to position and hold the rotameter in place, while the upper jig in (a) was used to align the face of the rotameter in (b) to coincide with the radial plane of the HDVS.

3.2 Numerical Setup

3.2.1 Governing Equations

The flow through the HDVS is assumed to be incompressible and steady state, and modelled using the Reynolds-averaged Navier-Stokes (RANS) equations (Versteeg and Malalasekera, 2007),

$$\nabla \cdot \overline{\mathbf{v}} = 0, \tag{3.2.1a}$$

$$\nabla \cdot (\overline{\mathbf{v}}\,\overline{\mathbf{v}}) = -\frac{1}{\rho}\nabla \overline{p} + \nu \nabla^2 \overline{\mathbf{v}} - \nabla \cdot \overline{\mathbf{v}'\mathbf{v}'}, \tag{3.2.1b}$$

where:

- $\overline{\mathbf{v}}$ is the time-averaged velocity,
- $\overline{p}$ is the time-averaged pressure,
- ρ is the density,
- ν is the kinematic viscosity and

- $\mathbf{v}'$ is the fluctuating component of the Reynolds decomposition of velocity.

The Reynolds stresses, $\overline{\mathbf{v}'\mathbf{v}'}$, represent the physical effect of the turbulent fluctuations around the mean flow and must be represented by a turbulence model in order to close the set of equations. RANS turbulence models fall into two categories: *i*) Reynolds stress models which model this term directly, either through algebraic models (second moment closure) or using a transport equation for the Reynolds stress, *ii*) Boussinesq models, in which the effects of the turbulence are represented by an additional turbulent viscosity. The turbulent viscosity in turn is derived from solving transport equations for turbulent quantities such as k (turbulent kinetic energy), ϵ (turbulent dissipation rate) and/or ω (turbulent frequency). Detailed derivation of the RANS turbulence equations is covered in Wilcox (2006); the turbulence models selected for this work are discussed in Section 3.2.2.

RANS equations represent a set of coupled, nonlinear transport equations, (3.2.1b), together with a constraint equation, (3.2.1a). Solution of these equations is not straightforward, but typically is achieved through discretisation of the equations using the finite volume approach, and application of the SIMPLE (steady state) or PISO (transient) algorithms, or variants of these. Both SIMPLE and PISO rely on the rearrangement of Eqns. (3.2.1) to form a Poisson equation for the pressure, p. Solution of the linked equations is then performed sequentially, as set out below (Jasak, 1996).

1. Solve Eqn. (3.2.1b) as a predictor step to derive $\mathbf{v}$.
2. Solve Poisson's equation for p.
3. Update the fluxes to satisfy continuity, Eqn. (3.2.1a).

Steps 1-3 are then repeated iteratively until the solution residuals (an estimate of the error in the matrix inversion) drop below specified values.

3.2.2 Turbulence Models

In order to assess the merits of different turbulence models, four were used to calculate the Reynolds stresses in the RANS equations; the first three being Boussinesq models, and the fourth calculating the Reynolds stresses directly, as set out below.

1. The realisable k-ϵ turbulence model, developed by Shih et al. (1995), has been shown to give superior results for flows involving rotation, separation and recirculation, compared with the standard k-ϵ modelling approach, developed by Jones and Launder (1972). Definitions of eddy viscosity and turbulent dissipation rate are similar to the standard k-ϵ model, although the proportionality constant (C_μ) varies depending on the normalised strain rate,

which prevents the Reynolds stresses becoming negative in regions of high strain rate. This ensures the model is realisable, i.e. the mean of the fluctuating quantities is zero, and improves the response of the model in regions of varying strain, e.g. in rotating and swirling flows (Jarman, 2011, p.158).

2. The Re-Normalisation Group (RNG) k-ϵ turbulence model, developed by Yakhot et al. (1992), accounts for the effects of smaller scales of motion. In the standard k-ϵ model, the eddy viscosity is determined from a single turbulent length scale (l), so the calculated turbulent diffusion is that which occurs only at the specified scale. In reality, all scales of motion contribute to the turbulent diffusion, which the RNG approach attempts to model through a modified form of the ϵ equation.
3. The k-ω turbulence model also applies the Boussinesq approximation, resulting in the turbulent kinetic energy transport equation, but differs in its handling of ϵ. Instead, it results in a transport equation for ω, rather than ϵ. Models of this type have been shown to be superior to the k-ϵ approach for boundary layer dominated flows, but do not perform as well as the k-ϵ models for free stream and wake conditions (Hoffmann and Chiang, 2000). In contrast, the k-ϵ models typically perform poorly in the near wall region (Launder and Sandham, 2002). In an attempt to benefit from the best features of both dissipation treatments, Menter and Esch (2001) blended the k-ϵ and k-ω methods to provide a more universal turbulence model: the k-ω shear stress transport (SST) turbulence model (Jarman, 2011).
4. The Launder-Reece-Rodi (LRR) Reynolds stress transport turbulence model, developed by Launder et al. (1975), computes the Reynolds stresses directly and therefore takes into account the directional effects of the Reynolds stress fields.

The initial values of k, ϵ and ω were calculated using Eqns. (3.2.2), taken from CFD Online (2014):

$$\mathsf{k} = \frac{3}{2}(v_{in}I)^2, \tag{3.2.2a}$$

$$\epsilon = C_\mu \frac{\mathsf{k}^{\frac{3}{2}}}{l}, \tag{3.2.2b}$$

$$\omega = \frac{\sqrt{\mathsf{k}}}{l}, \tag{3.2.2c}$$

where I is the initial turbulent intensity, estimated as $0.16Re_{in}^{-\frac{1}{8}}$, with $Re_{in} = \frac{v_{in}D_{in}}{\nu}$ being the Reynolds number of the inlet pipe. Initially, the Reynolds stress tensor was populated with $\frac{2}{3}$k in the diagonal terms and zero in the off-diagonal terms, where the initial values of k, ϵ and ω are presented in Table 3.1.

Table 3.1: Initial values of k, ϵ and ω used in turbulence models.

	k (m^2/s^2)	ϵ (m^2/s^3)	ω (-/s)
Initial values	0.005	2×10^{-7}	37

3.2.3 Wall Functions

The governing equations and turbulence model equations were not solved in the boundary layer, close to the walls, because it can be computationally expensive to do so and because the velocity profile in the boundary layer is well understood (Versteeg and Malalasekera, 2007). Hence wall functions were used to model the flow in the boundary layer, with the necessary constants obtained from Versteeg and Malalasekera (2007). The condition for employing wall functions is that the average y^+ value should be between 30 and 60 (Salim and Cheah, 2009). However the y^+ values calculated on the converged solutions, which are recorded in Appendix A.4 and summarised in Tables 3.4 and 3.6, show that the wall function condition was only satisfied for the meshes with anisotropic boundary layer(s).

3.2.4 Mesh

The above equations are solved in OpenFOAM® using the finite volume method (Versteeg and Malalasekera, 2007). In this methodology, the domain of interest is split up into numerous control volumes or cells, and the governing equations integrated over each cell to generate discretised, difference equations which can be evaluated by matrix inversion. The collection of cells is referred to as a mesh, and the process of constructing the mesh, known as meshing, is one of the most complex and time-consuming parts of the CFD process. Spatial derivatives are treated through the application of Gauss' theorem to convert them into fluxes on the cell faces, which are evaluated by interpolation from the cell centres, where the values are stored. This interpolation process, referred to as the differencing scheme, introduces numerical errors which are related to the cell shape and mesh structure, so its construction is of critical importance for the solution process.

In order to assess the merits of different mesh types and refinement, four different meshes were constructed using Pointwise®. The first was an unstructured isotropic mesh filled with tetrahedra, Figure 3.7, and the second was the same as the first, except with twice the number of nodes. An unstructured mesh structure was chosen because it is easier to create than a structured one and because of the complex internal geometry involved. However, simulations invariably take longer to converge on an unstructured mesh, because neighbouring cells usually have a higher non-orthogonality than on a structured mesh.

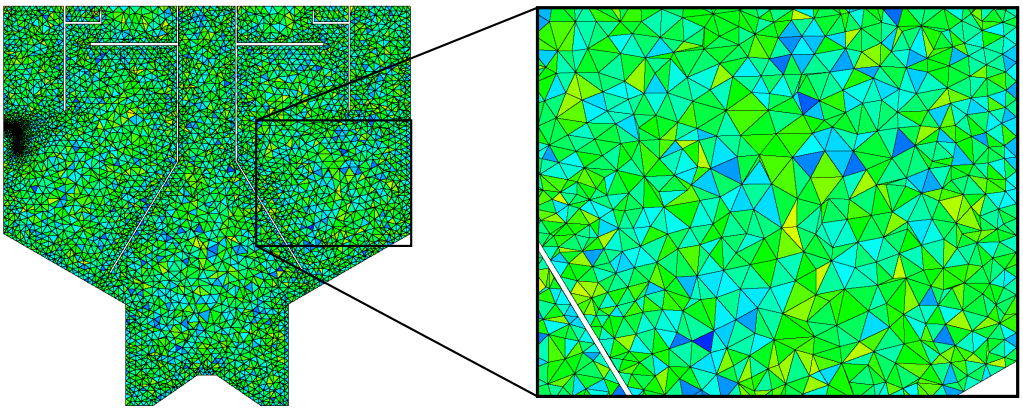


Figure 3.7: Unstructured isotropic mesh filled with tetrahedra.

Figure 3.8 illustrates the notation used in determining mesh non-orthogonality, where f represents the common face between two cells, with cell centres at c_1 and c_2, p_1 is the face centre, p_2 is the intersection of the face and the line c_1c_2, connecting the two cell centres, and **n** is the vector normal to the face. Non-orthogonality is defined as the angle, θ, between c_1c_2 and **n**.

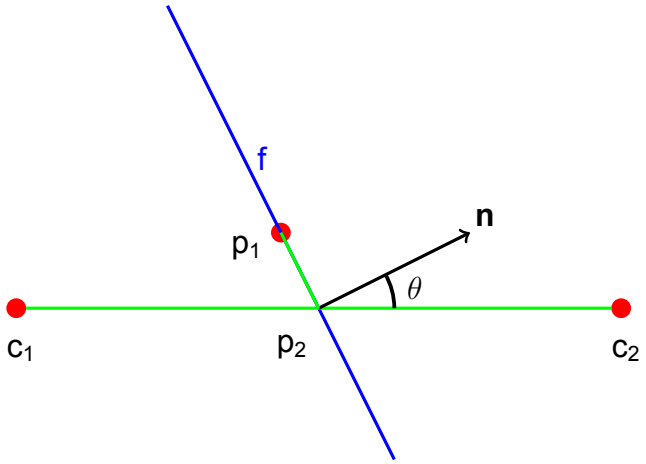


Figure 3.8: Mesh non-orthogonality notation.

The other two meshes were the same as the first, except with boundary layer(s) of anisotropic tetrahedra extruded from the walls (T-Rex® cells), Figure 3.9. These cells are combined to create structured hexahedral cells, but more importantly, the T-Rex cell heights can be set independently of the isotropic cell size, which is invaluable when using wall functions because they require the boundary layer to be set a certain distance from the wall.

The statistics for all four meshes were generated, using OpenFOAM®'s utility for checking mesh validity, and are recorded in Appendix A.3. The meshes were all approved and the number of cells and maximum non-orthogonality are summarised in Table 3.2. It is evident that one of the main advantages of using T-Rex cells is that the number of cells is significantly reduced.

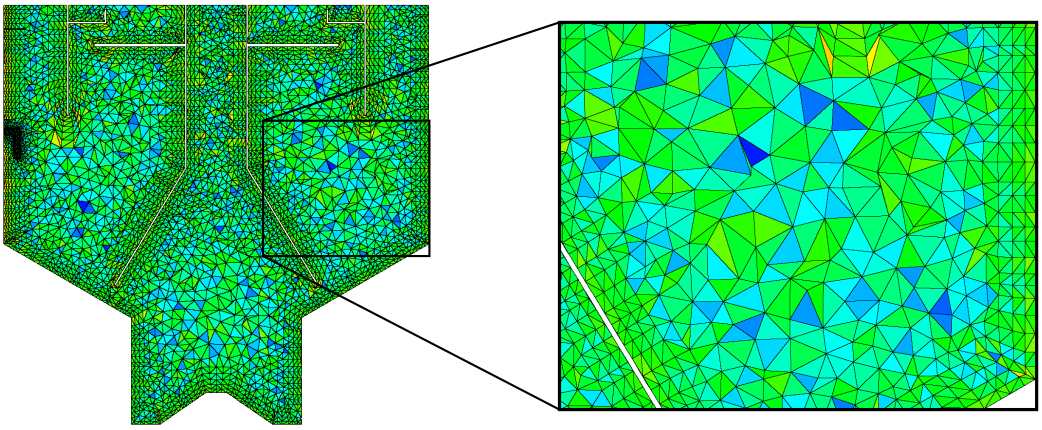


Figure 3.9: Unstructured isotropic mesh with T-Rex® cells at the walls.

Table 3.2: Mesh statistics.

	Isotropic mesh		Isotropic mesh with T-Rex boundary layer	
	Coarse	Fine	Coarse	Fine
Number of cells (million)	1.6	8.1	0.9	5.4
Max non-orthogonality	81°	84°	81°	85°

3.2.5 Software and Hardware

The governing equations, turbulence models and wall functions were implemented using OpenFOAM® (CFD Direct, 2017) and the simulations were run in parallel on an Intel Xeon E5-4620 CPU, using four cores running at 2.20 GHz, with 132 GB RAM. OpenFOAM® was used because it is open source software and, therefore, is publicly accessible. There are two advantages with this: the first is that it is free to use and the second is that anyone can inspect the source code (OpenSource, 2017). The first is especially beneficial when running simulations in parallel, because proprietary software companies often require a license for each core. The second is beneficial, because it allows the user to view the code, copy it, learn from it and alter it. In Chapter 4, the adjoint drift flux equations will be derived and coded and, as far as the author is aware, this has not been done before. As OpenFOAM® includes the source code for the adjoint Navier-Stokes equations, it will serve as the foundation for coding the adjoint drift flux equations.

3.2.6 Comparison Procedure

Simulations were run on each of the four meshes, using the realisable k-ϵ turbulence model, to determine the mesh that produced velocities closest to the experimental velocities. Four simulations were then run on that mesh, three using the remaining turbulence models and one with no turbulence model, to determine the model that most accurately matched the experimental results. Three performance indicators (Moriasi et al., 2007) were used to evaluate the relationship between the experimental and simulation tangential velocities:

1. Pearson's correlation coefficient, which ranges from -1 to 1, with a strong positive linear relationship indicated by the coefficient being closer to 1.
2. The percent bias (PBIAS), whose optimal value is zero and a very good performance rating is indicated by PBIAS < ±10.
3. Nash-Sutcliffe efficiency (NSE), which ranges from -∞ to 1, with a very good performance rating indicated by 0.75 < NSE < 1.

3.3 Results

The results of the mesh and turbulence model comparison studies are recorded in Appendices A.5 and A.6, respectively. Convergence was determined by inspecting the residuals of the state variables and the distribution of the pressure and velocity fields. The experimental and simulated velocities were compared at the locations described in Section 3.1.3, using the three performance indicators described above, and the simulation run times were also compared, because speed as well as accuracy is an important criteria when determining the most appropriate mesh and turbulence model to use.

3.3.1 Mesh Comparison Study

Appendices A.5.1 and A.5.2 show that the simulations on all four meshes reached a converged solution, because the residuals were < 0.01 and the magnitude of the velocity and pressure distributions were all reasonably small. The results of the mesh comparison study are given in Table 3.3, with the values in bold indicating the best performance rating for each indicator. It shows that the coarse isotropic mesh with T-Rex boundary layer performed better than the other meshes for all three performance indicators. Though none of the meshes performed very well according to the PBIAS or NSE indicators, all four meshes performed very well according to Pearson's correlation coefficient, indicating a strong linear correlation between the experimental and simulation data.

Table 3.3: Mesh comparison study: performance indicator results, with values in bold indicating the best performance rating.

	Isotropic mesh		Isotropic mesh with T-Rex boundary layer	
	Coarse	Fine	Coarse	Fine
Pearson's correlation coeff.	0.877	0.894	**0.904**	0.900
Percent bias	-61.79	-45.02	**-33.78**	-40.46
Nash-Sutcliffe efficiency	-1.61	-0.52	**0.10**	-0.25

The velocity profiles of the experimental and simulated velocities are recorded in Appendix A.5.3, two of which are presented here to highlight the general trend of the results. Figure 3.10 shows that the simulated velocities on all four meshes were greater than the experimental velocities, though they matched the profile of the experimental velocities fairly well, as demonstrated by Pearson's correlation coefficient being close to 1 in Table 3.3. The meshes with the anisotropic boundary layer(s) produced results that more closely corresponded to the experimental values than the meshes without boundary layer(s), and the coarse mesh with the anisotropic boundary layer(s) produced the most accurate results of all, as demonstrated by the bolded values in Table 3.3.

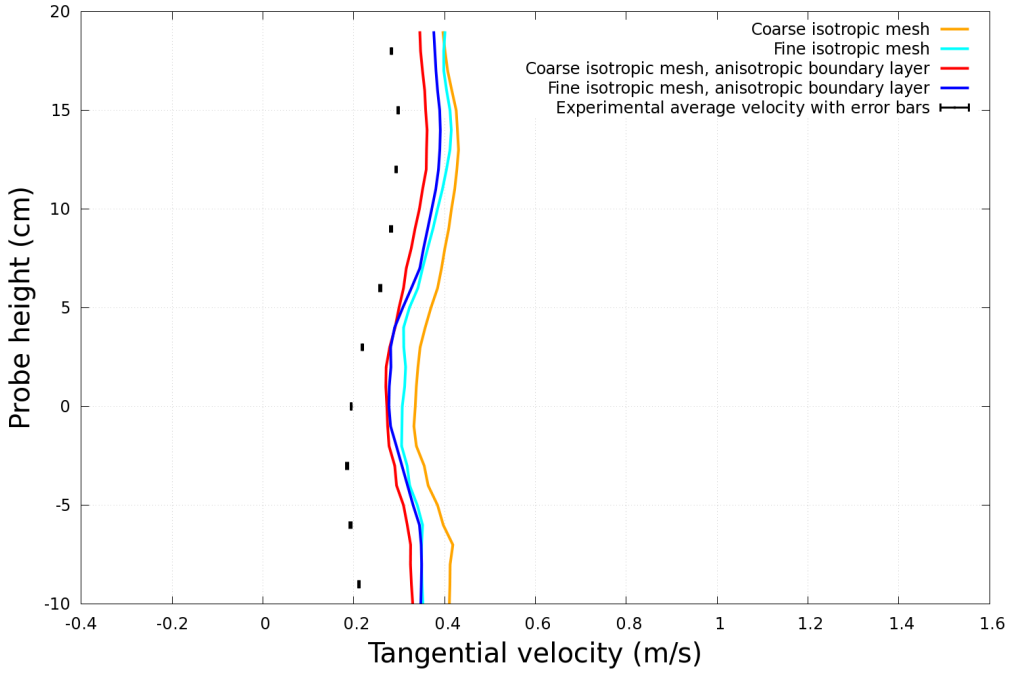


Figure 3.10: Mesh comparison study: velocity profile at second quadrant, middle ring.

Figure 3.11 shows that the simulated velocities overestimated the experimental velocities and that the coarse mesh with the anisotropic boundary layer(s) again

produced the most accurate results, however at this location the meshes without an anisotropic boundary layer were more erratic. The velocity readings were taken only 1 cm from the cylinder wall and the most probable explanation for the erratic results is the lack of an anisotropic boundary layer at the wall, resulting in average y^+ values < 30, Table 3.4.

Table 3.4: Mesh comparison study: average y^+ values and convergence times.

	Isotropic mesh		Isotropic mesh with T-Rex boundary layer	
	Coarse	Fine	Coarse	Fine
Average y^+ value, cylinder (-)	16	9	51	37
Time to convergence (hours)	33	154	7	105

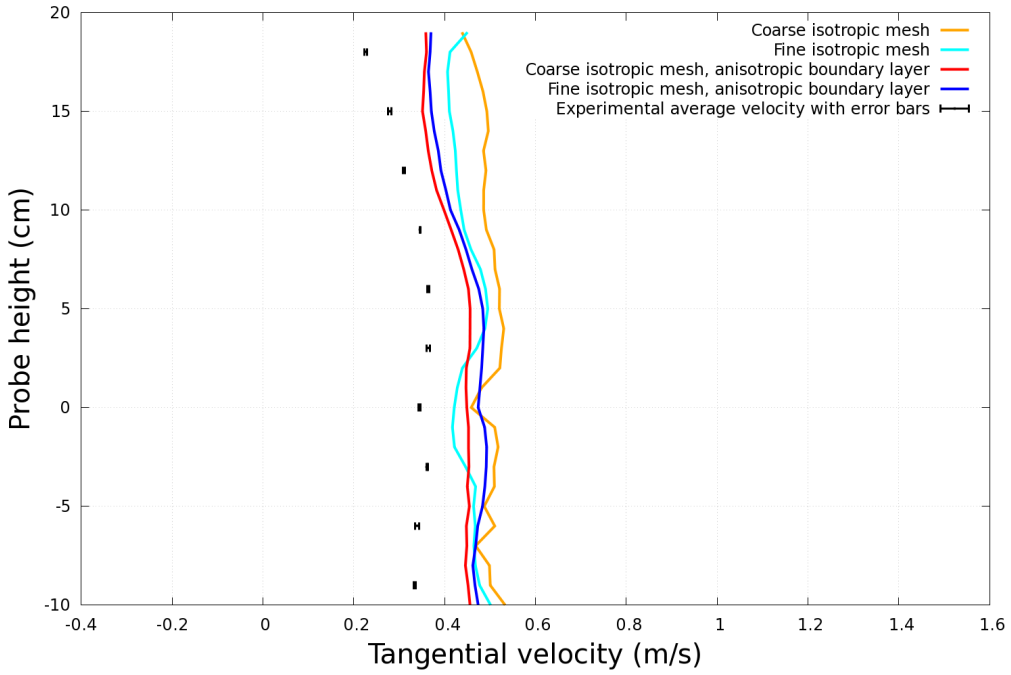


Figure 3.11: Mesh comparison study: velocity profile at third quadrant, outer ring.

The time to convergence is also presented in Table 3.4 and it is noticeable that the simulation converged fastest on the coarse mesh with the anisotropic boundary layer(s). Therefore, the fastest and most accurate results were produced on this mesh and, consequently, the turbulence model comparison study was performed on the coarse mesh with the anisotropic boundary layer(s).

3.3.2 Turbulence Model Comparison Study

As with the mesh comparison study, convergence was determined by inspecting the residuals of the state variables and the distribution of the pressure and

velocity fields, Appendices A.6.1 and A.6.2, respectively. It is noticeable that the simulation without a turbulence model did not converge to the same degree as the simulations with a turbulence model, and that the distribution of its velocity vectors were more dispersed than for the other simulations. However, as the residuals were relatively unchanged for several thousand iterations and the magnitude of the velocity and pressure distributions were reasonably small, all four simulations reached a converged solution.

The results of the turbulence model comparison study are given in Table 3.5, with the values in bold indicating the best performance rating for each indicator. It is noticeable that for all five models, including the laminar model, Pearson's correlation coefficient is close to 1, indicating a strong linear correlation between the experimental and simulation data. Only the LRR turbulence model performed well for the other two performance indicators, according to the criteria given in Section 3.2.6.

Table 3.5: Turbulence model comparison study: performance indicator results, with values in bold indicating the best performance rating.

	Real. k-ϵ	RNG k-ϵ	k-ω SST	LRR	Laminar
Pearson's corr. coeff.	**0.904**	0.901	0.894	0.890	0.845
Percent bias	-33.78	-34.91	-36.49	**-7.07**	-81.24
Nash-Sutcliffe effic.	0.10	0.05	-0.02	**0.69**	-3.39

The velocity profiles of the experimental and simulated velocities are recorded in Appendix A.6.3, one of which is presented here to draw attention to several noteworthy features from the results. Figure 3.12 shows that the LRR turbulence model most closely corresponded to the experimental velocities, the eddy viscosity models produced similar results that overestimated the experimental velocities, and the laminar model most overestimated the experimental velocities, as indicated by the values in Table 3.5. However, they all matched the profile of the experimental velocities reasonably well, as demonstrated by Pearson's correlation coefficient in Table 3.5.

Table 3.6 shows the average y^+ values for the turbulence models and the time to convergence for all simulations. The average y^+ values are all in the required range, between 30 and 60, and the time to convergence is significantly shorter for the laminar model than for the simulations with turbulence models.

Table 3.6: Turbulence model comparison study: average y^+ values and convergence times.

	Real. k-ϵ	RNG k-ϵ	k-ω SST	LRR	Laminar
Average y^+ value, cyl.	51	51	52	41	-
Time to conv. (hours)	7	30	9	15	1

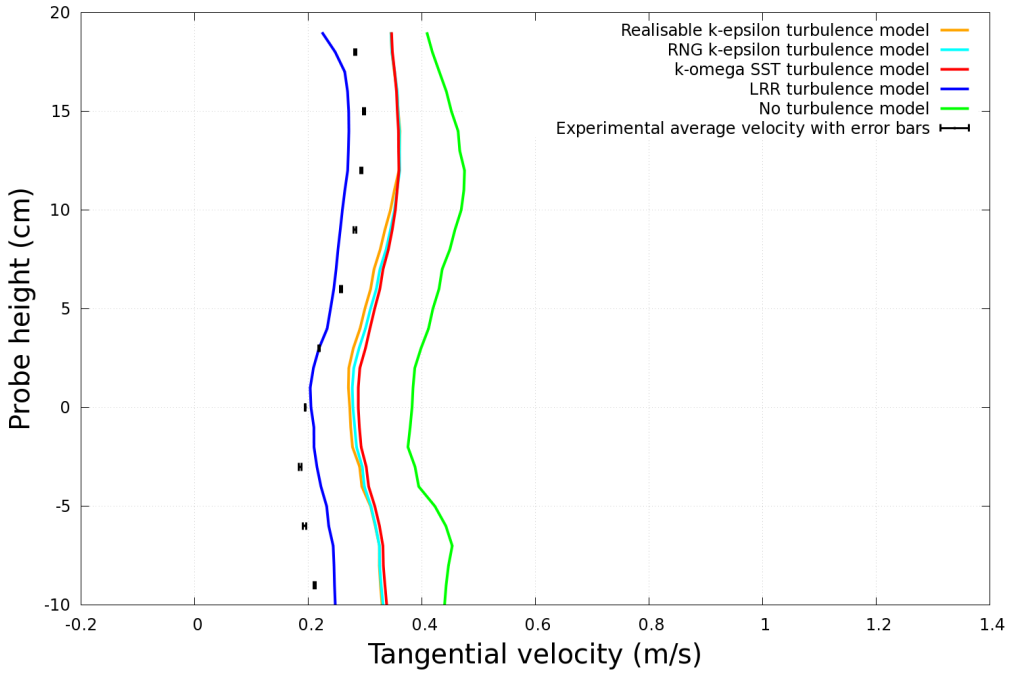


Figure 3.12: Turbulence model comparison study: velocity profile at second quadrant, middle ring.

3.3.3 Reynolds Number

The Reynolds number, defined as,

$$Re = \frac{vL}{\nu}, \tag{3.3.1}$$

where L is a characteristic length, is used to indicate when flow is turbulent. Using diameter as their characteristic length, Reynolds numbers were calculated in the inlet pipe and HDVS cylinder and suggest that the flow was turbulent, Table 3.7. However, because the inlet velocity was not scaled using its Reynolds number, the Reynolds numbers for the full-size geometry would be an order of magnitude greater than for the model.

Table 3.7: Reynolds numbers for the model HDVS.

	Inlet pipe	HDVS cylinder
Reynolds number (-)	38,500	144,000

3.3.4 Error Analysis

There are two types of error associated with conducting an experiment: accuracy and precision (Pugh and Winslow, 1966). The accuracy error depends on the uncertainty of the experimental setup, in this case on the flow rate and position and orientation of the rotameter, which are estimated in Table 3.8.

Table 3.8: Uncertainty errors.

	Uncertainty error
Flow rate	$\pm$ 0.02 l/s
Position of rotameter	$\pm$ 3 mm
Orientation of rotameter	$\pm$ 5°

As explained in Section 3.1.4, the rotameter readings were mean values. The standard error of the mean is given by,

$$\sigma_{\overline{f}} = \sqrt{\frac{\sum(f_i^2) - n\overline{f}^2}{n(n-1)}}, \tag{3.3.2}$$

where f_i are the rotameter frequency readings, n is the number of readings and $\overline{f}$ is the mean value of the readings, taken at each location. The precision error is defined as the ratio of the standard error of the mean to the mean value of the readings (LAHC, 2017),

$$\frac{\sigma_{\overline{f}}}{\overline{f}}, \tag{3.3.3}$$

which is less than 4% for this experiment and demonstrates reasonably consistent readings.

3.4 Conclusions

Of the four meshes used, the two with the anisotropic boundary layer(s) converged more quickly and produced results that more closely corresponded with the experimental values than those without boundary layer(s). This is probably due to the wall function condition only being satisfied for the two meshes with the anisotropic boundary layer(s). Of the two meshes with the anisotropic boundary layer(s), the coarse mesh produced more accurate results than the fine mesh, which is unexpected because usually a finer mesh produces more accurate results. However, the boundary layer height was greater for the coarse mesh, which was more suitable for the wall functions employed there. As a result of this study and learning that hexahedral cells produce more accurate results than tetrahedral cells, it is recommended to use a structured mesh in Chapter 5, the optimisation study.

The LRR turbulence model produced the best fit to the experimental measurements, the eddy viscosity models overestimated the experimental velocities, and the simulation without a turbulence model produced the worst fit to the experimental values. However, the latter consistently overestimated the experimental values and took significantly less time to converge than the other simulations. In Chapter 5, where it is necessary to run several CFD simulations, this would be a significant benefit especially if only the velocity profile is required.

Chapter 4

Theory Development and Methodology

The aim of this chapter is to present the theory and methodology developed in this research. A topological shape modifier is incorporated into the drift flux equations, and the continuous adjoint method is used to derive the adjoint drift flux equations and boundary conditions. The topological sensitivity is computed to calculate porosity, and a procedure is presented to optimise the design. Equations are specified for ducted flows, using an objective function for HDVS, and are coded using OpenFOAM®.

4.1 The Optimisation Problem

If the performance of a device is measured by an objective function, J, and the residuals of the primal (flow) equations are given by $\mathcal{R}$, the optimisation problem can be stated as,

$$\text{optimise } J(\mathsf{x},\mathsf{y}) \text{ subject to } \mathcal{R}(\mathsf{x},\mathsf{y}) = 0, \tag{4.1.1}$$

where x are the design parameters and y are the primal variables. It can then be formulated as,

$$\mathscr{L} = J + \int_{\Omega} \lambda \mathcal{R} \, d\Omega, \tag{4.1.2}$$

where $\mathscr{L}$ is the Lagrange function, λ are the Lagrange multipliers (also referred to as the adjoint variables) and Ω is the flow domain. In this case, the primal equations are the steady state drift flux equations, Section 2.3.3, with the capillary force taken to be zero (Brennan, 2001) and a Darcy term included in the mixture-momentum equation, Section 2.4.2. They are rearranged in terms of their

residuals, $\mathcal{R} = (R_1, R_2, R_3, R_4, R_5)^T$, as follows:

$$(R_1, R_2, R_3)^T = (\mathbf{v}_m \cdot \nabla)(\rho_m \mathbf{v}_m) + \nabla(\rho_m p_m) - \nabla \cdot \left(2\mu_m \mathsf{D}(\mathbf{v}_m)\right) + \nabla \cdot \left(\frac{\alpha}{1-\alpha}\frac{\rho_c \rho_d}{\rho_m}\mathbf{v}_{dj}\mathbf{v}_{dj}\right) - \rho_m \mathbf{g} + \aleph \rho_m \mathbf{v}_m, \tag{4.1.3a}$$

$$R_4 = -\nabla \cdot (\rho_m \mathbf{v}_m), \tag{4.1.3b}$$

$$R_5 = \nabla \cdot (\alpha \mathbf{v}_m) + \nabla \cdot \left(\frac{\alpha \rho_c}{\rho_m}\mathbf{v}_{dj}\right) - \nabla \cdot K \nabla \alpha. \tag{4.1.3c}$$

The variation of the Lagrange function with respect to the primal variables, $(\mathbf{v}_m, p_m, \alpha)$, and the design parameter, $\aleph$, is,

$$\delta \mathscr{L} = \delta_{\mathbf{v}_m} \mathscr{L} + \delta_{p_m} \mathscr{L} + \delta_\alpha \mathscr{L} + \delta_\aleph \mathscr{L}, \tag{4.1.4}$$

where, for example, $\delta_\alpha \mathscr{L} = \mathscr{L}(\alpha + \delta\alpha) - \mathscr{L}(\alpha)$. We choose the adjoint variables, $(\mathbf{u}, q, \beta) = (u_1, u_2, u_3, q, \beta)$, so that the variation with respect to the primal variables vanishes, i.e.

$$\delta_{\mathbf{v}_m} \mathscr{L} + \delta_{p_m} \mathscr{L} + \delta_\alpha \mathscr{L} = 0, \tag{4.1.5}$$

and the Lagrange function now varies only with respect to the design parameter,

$$\delta \mathscr{L} = \delta_\aleph \mathscr{L} = \delta_\aleph J + \int_\Omega (\mathbf{u}, q, \beta) \delta_\aleph \mathcal{R} \, d\Omega. \tag{4.1.6}$$

Although $\mathbf{u}$ and q are called the adjoint velocity and adjoint pressure, respectively, they should not be interpreted as a velocity or a pressure in the physical sense (Nilsson, 2014). The names are used, because *a*) the number of primal and adjoint variables is the same, and *b*) the primal and adjoint equations are similar, which leads to a similar solution procedure.

4.1.1 Derivation of the Adjoint Drift Flux Equations

The adjoint drift flux equations are derived by substituting Eqn. (4.1.2) into Eqn. (4.1.5), giving,

$$\delta_{\mathbf{v}_m} J + \delta_{p_m} J + \delta_\alpha J + \int_\Omega (\mathbf{u}, q, \beta) \delta_{\mathbf{v}_m} \mathcal{R} \, d\Omega + \int_\Omega (\mathbf{u}, q, \beta) \delta_{p_m} \mathcal{R} \, d\Omega + \int_\Omega (\mathbf{u}, q, \beta) \delta_\alpha \mathcal{R} \, d\Omega = 0, \tag{4.1.7}$$

which can be expanded to,

$$\begin{aligned}\delta_{\mathbf{v}_m} J + \delta_{p_m} J + \delta_\alpha J + \int_\Omega d\Omega\, \mathbf{u} \cdot \delta_{\mathbf{v}_m}(R_1, R_2, R_3)^T + \int_\Omega d\Omega\, q\delta_{\mathbf{v}_m} R_4 + \int_\Omega d\Omega\, \beta\delta_{\mathbf{v}_m} R_5 \\ + \int_\Omega d\Omega\, \mathbf{u} \cdot \delta_{p_m}(R_1, R_2, R_3)^T + \int_\Omega d\Omega\, q\delta_{p_m} R_4 + \int_\Omega d\Omega\, \beta\delta_{p_m} R_5 \\ + \int_\Omega d\Omega\, \mathbf{u} \cdot \delta_\alpha(R_1, R_2, R_3)^T + \int_\Omega d\Omega\, q\delta_\alpha R_4 + \int_\Omega d\Omega\, \beta\delta_\alpha R_5 = 0. \quad (4.1.8)\end{aligned}$$

The variation of $(R_1, R_2, R_3)^T$ with respect to α is calculated as,

$$\begin{aligned}\delta_\alpha(R_1, R_2, R_3)^T &= \delta_\alpha\Bigg((\mathbf{v}_m \cdot \nabla)(\rho_m \mathbf{v}_m) + \nabla(\rho_m p_m) - \nabla \cdot \big(2\mu_m \mathsf{D}(\mathbf{v}_m)\big) \\ &\qquad + \nabla \cdot \left(\frac{\alpha}{1-\alpha}\frac{\rho_c \rho_d}{\rho_m}\mathbf{v}_{dj}\mathbf{v}_{dj}\right) - \rho_m \mathbf{g} + \aleph \rho_m \mathbf{v}_m\Bigg) \\ &= \delta_\alpha\big((\mathbf{v}_m \cdot \nabla)(\rho_m \mathbf{v}_m)\big) + \delta_\alpha \nabla(\rho_m p_m) + \delta_\alpha \nabla \cdot \mathsf{A} + \delta_\alpha\big(\rho_m(\aleph \mathbf{v}_m - \mathbf{g})\big), \quad (4.1.9)\end{aligned}$$

where

$$\mathsf{A} = \frac{\alpha}{1-\alpha}\frac{\rho_c \rho_d}{\rho_m}\mathbf{v}_{dj}\mathbf{v}_{dj}, \quad (4.1.10)$$

$$\begin{aligned}\rho_m &= \alpha\rho_d + (1-\alpha)\rho_c \\ &= (1-\alpha)\rho_c\left(\frac{\alpha\rho_d}{(1-\alpha)\rho_c} + 1\right) \\ &= (1-\alpha)\rho_c\left(1 + \frac{\alpha}{1-\alpha}\frac{\rho_d}{\rho_c}\right) \quad (4.1.11)\end{aligned}$$

and the variation of the mixture viscosity, μ_m, has been neglected. Substituting Eqn. (4.1.11) into Eqn. (4.1.10) and rewriting the parentheses as binomial expansions,

$$\begin{aligned}\mathsf{A}(\alpha) &= \alpha(1-\alpha)^{-1}\rho_c\rho_d\frac{(1-\alpha)^{-1}}{\rho_c}\left(1 + \frac{\alpha}{1-\alpha}\frac{\rho_d}{\rho_c}\right)^{-1}\mathbf{v}_{dj}\mathbf{v}_{dj} \\ &= \alpha(1-\alpha)^{-2}\rho_d\left(1 + \frac{\alpha}{1-\alpha}\frac{\rho_d}{\rho_c}\right)^{-1}\mathbf{v}_{dj}\mathbf{v}_{dj} \\ &= \alpha(1 + 2\alpha + \cdots)\rho_d\left(1 - \frac{\alpha}{1-\alpha}\frac{\rho_d}{\rho_c} + \cdots\right)\mathbf{v}_{dj}\mathbf{v}_{dj} \\ &= \alpha\rho_d\left(1 + \alpha\left(2 - \frac{1}{1-\alpha}\frac{\rho_d}{\rho_c}\right) + \cdots\right)\mathbf{v}_{dj}\mathbf{v}_{dj}. \quad (4.1.12)\end{aligned}$$

As $\alpha \ll 1$ and $\rho_d \approx 2\rho_c \implies \left|2 - \frac{1}{1-\alpha}\frac{\rho_d}{\rho_c}\right| < 1$ and ignoring terms containing squared and higher powers of α,

$$\mathsf{A}(\alpha) \approx \alpha\rho_d \mathbf{v}_{dj}\mathbf{v}_{dj}. \quad (4.1.13)$$

In this formulation $\mathbf{v}_{dj}$ is modelled using Eqn. (2.2.6),

$$\begin{aligned}\implies \mathsf{A}(\alpha+\delta\alpha) &= (\alpha+\delta\alpha)\rho_d\mathbf{v}_0\mathbf{v}_0 e^{-2k_1(\alpha+\delta\alpha)} \\ &= (\alpha+\delta\alpha)\rho_d\mathbf{v}_0\mathbf{v}_0 e^{-2k_1\alpha}e^{-2k_1\delta\alpha} \\ &= (\alpha+\delta\alpha)\rho_d\mathbf{v}_{dj}\mathbf{v}_{dj}(1-2k_1\delta\alpha+\cdots),\end{aligned} \tag{4.1.14}$$

where $e^{-2k_1\delta\alpha}$ is rewritten as a Taylor's series expansion and $k_1 = k\ln 10$. Ignoring terms containing squared and higher powers of $\delta\alpha$,

$$\begin{aligned}\delta_\alpha\nabla\cdot\mathsf{A} &= \nabla\cdot\mathsf{A}(\alpha+\delta\alpha) - \nabla\cdot\mathsf{A}(\alpha) \\ &\approx -\nabla\cdot\big(\delta\alpha\rho_d\mathbf{v}_{dj}\mathbf{v}_{dj}(2\alpha k_1-1)\big),\end{aligned} \tag{4.1.15}$$

$$\begin{aligned}\implies \delta_\alpha(R_1,R_2,R_3)^T &\approx (\mathbf{v}_m\cdot\nabla)\big(\delta\alpha(\rho_d-\rho_c)\mathbf{v}_m\big) + \nabla\big(\delta\alpha(\rho_d-\rho_c)p_m\big) \\ &\quad - \nabla\cdot\big(\delta\alpha\rho_d\mathbf{v}_{dj}\mathbf{v}_{dj}(2\alpha k_1-1)\big) + \delta\alpha(\rho_d-\rho_c)(\aleph\mathbf{v}_m-\mathbf{g}) \\ &\approx (\rho_d-\rho_c)\big((\mathbf{v}_m\cdot\nabla)(\delta\alpha\mathbf{v}_m) + \nabla(\delta\alpha p_m) + \delta\alpha(\aleph\mathbf{v}_m-\mathbf{g})\big) \\ &\quad - \nabla\cdot\big(\delta\alpha\rho_d\mathbf{v}_{dj}\mathbf{v}_{dj}(2\alpha k_1-1)\big).\end{aligned} \tag{4.1.16}$$

Similarly, the variation of R_5 with respect to α is calculated as,

$$\begin{aligned}\delta_\alpha R_5 &= \delta_\alpha\left(\nabla\cdot(\alpha\mathbf{v}_m) + \nabla\cdot\left(\frac{\alpha\rho_c}{\rho_m}\mathbf{v}_{dj}\right) - \nabla\cdot K\nabla\alpha\right) \\ &= \delta_\alpha\nabla\cdot(\alpha\mathbf{v}_m) + \delta_\alpha\nabla\cdot\mathsf{B} - \delta_\alpha(\nabla\cdot K\nabla\alpha),\end{aligned} \tag{4.1.17}$$

where

$$\mathsf{B} = \frac{\alpha\rho_c}{\rho_m}\mathbf{v}_{dj}. \tag{4.1.18}$$

Substituting Eqn. (4.1.11) into Eqn. (4.1.18) and rewriting the parentheses as binomial expansions,

$$\begin{aligned}\mathsf{B}(\alpha) &= \alpha\rho_c\frac{(1-\alpha)^{-1}}{\rho_c}\left(1+\frac{\alpha}{1-\alpha}\frac{\rho_d}{\rho_c}\right)^{-1}\mathbf{v}_{dj} \\ &= \alpha(1-\alpha)^{-1}\left(1+\frac{\alpha}{1-\alpha}\frac{\rho_d}{\rho_c}\right)^{-1}\mathbf{v}_{dj} \\ &= \alpha(1+\alpha+\cdots)\left(1-\frac{\alpha}{1-\alpha}\frac{\rho_d}{\rho_c}+\cdots\right)\mathbf{v}_{dj} \\ &= \alpha\left(1+\alpha\left(1-\frac{1}{1-\alpha}\frac{\rho_d}{\rho_c}\right)+\cdots\right)\mathbf{v}_{dj}.\end{aligned} \tag{4.1.19}$$

As $\alpha \ll 1$ and $\rho_d \approx 2\rho_c \implies \left|1-\frac{1}{1-\alpha}\frac{\rho_d}{\rho_c}\right| < 2$ and ignoring terms containing squared and higher powers of α,

$$\mathsf{B}(\alpha) \approx \alpha\mathbf{v}_{dj}, \tag{4.1.20}$$

$$\begin{aligned}\implies \mathsf{B}(\alpha+\delta\alpha) &= (\alpha+\delta\alpha)\mathbf{v}_0 e^{-k_1(\alpha+\delta\alpha)}\\ &= (\alpha+\delta\alpha)\mathbf{v}_0 e^{-k_1\alpha}e^{-k_1\delta\alpha}\\ &= (\alpha+\delta\alpha)\mathbf{v}_{dj}(1-k_1\delta\alpha+\cdots),\end{aligned} \tag{4.1.21}$$

where $e^{-k_1\delta\alpha}$ is rewritten as a Taylor's series expansion. Ignoring terms containing squared and higher powers of $\delta\alpha$,

$$\begin{aligned}\delta_\alpha(\nabla\cdot\mathsf{B}) &= \nabla\cdot\mathsf{B}(\alpha+\delta\alpha)-\nabla\cdot\mathsf{B}(\alpha)\\ &\approx -\nabla\cdot\big(\delta\alpha\mathbf{v}_{dj}(\alpha k_1-1)\big),\end{aligned} \tag{4.1.22}$$

$$\implies \delta_\alpha R_5 \approx \nabla\cdot(\delta\alpha\mathbf{v}_m)-\nabla\cdot\big(\delta\alpha\mathbf{v}_{dj}(\alpha k_1-1)\big)-\nabla\cdot K\nabla\delta\alpha. \tag{4.1.23}$$

The variation of $(R_1,R_2,R_3,R_4,R_5)^T$ with respect to the primal variables can now be summarised as:

$$\begin{aligned}\delta_{\mathbf{v}_m}(R_1,R_2,R_3)^T &= (\delta\mathbf{v}_m\cdot\nabla)(\rho_m\mathbf{v}_m)+(\mathbf{v}_m\cdot\nabla)(\rho_m\delta\mathbf{v}_m)-\nabla\cdot\big(2\mu_m\mathsf{D}(\delta\mathbf{v}_m)\big)\\ &\quad+\aleph\rho_m\delta\mathbf{v}_m, && \text{(4.1.24a)}\\ \delta_{\mathbf{v}_m}R_4 &= -\nabla\cdot(\rho_m\delta\mathbf{v}_m), && \text{(4.1.24b)}\\ \delta_{\mathbf{v}_m}R_5 &= \nabla\cdot(\alpha\delta\mathbf{v}_m), && \text{(4.1.24c)}\\ \delta_{p_m}(R_1,R_2,R_3)^T &= \nabla(\rho_m\delta p_m), && \text{(4.1.24d)}\\ \delta_{p_m}R_4 &= 0, && \text{(4.1.24e)}\\ \delta_{p_m}R_5 &= 0, && \text{(4.1.24f)}\\ \delta_\alpha(R_1,R_2,R_3)^T &= (\rho_d-\rho_c)\big((\mathbf{v}_m\cdot\nabla)(\delta\alpha\mathbf{v}_m)+\nabla(\delta\alpha p_m)+\delta\alpha(\aleph\mathbf{v}_m-\mathbf{g})\big)\\ &\quad-\nabla\cdot\big(\delta\alpha\rho_d\mathbf{v}_{dj}\mathbf{v}_{dj}(2\alpha k_1-1)\big), && \text{(4.1.24g)}\\ \delta_\alpha R_4 &= -(\rho_d-\rho_c)\nabla\cdot(\delta\alpha\mathbf{v}_m), && \text{(4.1.24h)}\\ \delta_\alpha R_5 &= \nabla\cdot(\delta\alpha\mathbf{v}_m)-\nabla\cdot\big(\delta\alpha\mathbf{v}_{dj}(\alpha k_1-1)\big)-\nabla\cdot K\nabla\delta\alpha, && \text{(4.1.24i)}\end{aligned}$$

where $\mathbf{v}_{dj}=\mathbf{v}_0 10^{-k\alpha}$ and $k_1=k\ln 10$, $\mathbf{v}_0$ and k being constants (Section 2.3.3). Here, the variation of the mixture viscosity, μ_m, has been neglected. This is correct only for laminar, single phase flow regimes. For turbulent flows, neglecting this variation constitutes a common approximation, known as *frozen turbulence*, which has been extended here to multiphase flow and is called *frozen viscosity*. With

these variations, Eqn. (4.1.8) now reads,

$$
\begin{aligned}
&\delta_{\mathbf{v}_m} J + \delta_{p_m} J + \delta_\alpha J \\
&+ \int_\Omega d\Omega\, \mathbf{u} \cdot \Big((\delta\mathbf{v}_m \cdot \nabla)(\rho_m \mathbf{v}_m) + (\mathbf{v}_m \cdot \nabla)(\rho_m \delta\mathbf{v}_m) - \nabla \cdot \big(2\mu_m \mathsf{D}(\delta\mathbf{v}_m)\big) + \aleph \rho_m \delta\mathbf{v}_m \Big) \\
&\quad - \int_\Omega d\Omega\, q \nabla \cdot (\rho_m \delta\mathbf{v}_m) + \int_\Omega d\Omega\, \beta \nabla \cdot (\alpha \delta\mathbf{v}_m) + \int_\Omega d\Omega\, \mathbf{u} \cdot \nabla(\rho_m \delta p_m) \\
&\quad - \int_\Omega d\Omega\, \mathbf{u} \cdot \Big(\nabla \cdot \big(\delta\alpha \rho_d \mathbf{v}_{dj} \mathbf{v}_{dj} (2\alpha k_1 - 1) \big) \Big) \\
&\quad + (\rho_d - \rho_c) \int_\Omega d\Omega\, \mathbf{u} \cdot \big((\mathbf{v}_m \cdot \nabla)(\delta\alpha \mathbf{v}_m) + \nabla(\delta\alpha p_m) + \delta\alpha(\aleph \mathbf{v}_m - \mathbf{g}) \big) \\
&\quad - (\rho_d - \rho_c) \int_\Omega d\Omega\, q \nabla \cdot (\delta\alpha \mathbf{v}_m) \\
&\quad + \int_\Omega d\Omega\, \beta \Big(\nabla \cdot (\delta\alpha \mathbf{v}_m) - \nabla \cdot \big(\delta\alpha \mathbf{v}_{dj} (\alpha k_1 - 1) \big) - \nabla \cdot K \nabla \delta\alpha \Big) = 0. \quad (4.1.25)
\end{aligned}
$$

Decomposing the objective function into contributions from the boundary, Γ, and interior, Ω, of the domain,

$$J = \int_\Gamma J_\Gamma \, d\Gamma + \int_\Omega J_\Omega \, d\Omega, \tag{4.1.26}$$

the terms in Eqn. (4.1.25) can be rewritten as follows. The variations of the objective function can be written as,

$$\delta_{\mathbf{v}_m} J = \int_\Gamma d\Gamma \, \frac{\partial J_\Gamma}{\partial \mathbf{v}_m} \cdot \delta\mathbf{v}_m + \int_\Omega d\Omega \, \frac{\partial J_\Omega}{\partial \mathbf{v}_m} \cdot \delta\mathbf{v}_m, \tag{4.1.27}$$

$$\delta_{p_m} J = \int_\Gamma d\Gamma \, \frac{\partial J_\Gamma}{\partial p_m} \delta p_m + \int_\Omega d\Omega \, \frac{\partial J_\Omega}{\partial p_m} \delta p_m \tag{4.1.28}$$

and

$$\delta_\alpha J = \int_\Gamma d\Gamma \, \frac{\partial J_\Gamma}{\partial \alpha} \delta\alpha + \int_\Omega d\Omega \, \frac{\partial J_\Omega}{\partial \alpha} \delta\alpha. \tag{4.1.29}$$

Applying the product rule, divergence theorem and continuity equation, and using the Einstein notation for clarity, the terms containing $\mathbf{u}$, $\mathbf{v}_m$ and ∇ can be written

as,

$$\begin{aligned}\int_\Omega d\Omega\, \mathbf{u}\cdot(\delta\mathbf{v}_m\cdot\nabla)(\rho_m\mathbf{v}_m) &= \int_\Omega d\Omega\, u_k\delta v_{mi}\frac{\partial}{\partial x_i}(\rho_{mk}v_{mk})\\ &= \int_\Omega d\Omega\frac{\partial}{\partial x_i}(u_k\rho_{mk}v_{mk}\delta v_{mi}) - \int_\Omega d\Omega\,\rho_{mk}v_{mk}\frac{\partial}{\partial x_i}(u_k\delta v_{mi})\\ &= \int_\Gamma d\Gamma\,\mathsf{n}_i u_k\rho_{mk}v_{mk}\delta v_{mi} - \int_\Omega d\Omega\,\rho_{mk}v_{mk}\delta v_{mi}\frac{\partial u_k}{\partial x_i}\\ &\quad - \int_\Omega d\Omega\,\rho_{mk}v_{mk}u_k\frac{\partial\delta v_{mi}}{\partial x_i}\\ &= \int_\Gamma d\Gamma\,\mathbf{n}(\mathbf{u}\cdot\rho_m\mathbf{v}_m)\cdot\delta\mathbf{v}_m - \int_\Omega d\Omega\,\nabla\mathbf{u}\cdot(\rho_m\mathbf{v}_m)\cdot\delta\mathbf{v}_m,\end{aligned} \tag{4.1.30}$$

$$\begin{aligned}\int_\Omega d\Omega\,\mathbf{u}\cdot(\mathbf{v}_m\cdot\nabla)\rho_m\delta\mathbf{v}_m &= \int_\Omega d\Omega\, u_k v_{mi}\frac{\partial}{\partial x_i}(\rho_{mk}\delta v_{mk})\\ &= \int_\Omega d\Omega\,\frac{\partial}{\partial x_i}(u_k v_{mi}\rho_{mk}\delta v_{mk}) - \int_\Omega d\Omega\,\rho_{mk}\delta v_{mk}\frac{\partial}{\partial x_i}(u_k v_{mi})\\ &= \int_\Gamma d\Gamma\,\mathsf{n}_i u_k v_{mi}\rho_{mk}\delta v_{mk} - \int_\Omega d\Omega\,\rho_{mk}\delta v_{mk}v_{mi}\frac{\partial u_k}{\partial x_i}\\ &\quad - \int_\Omega d\Omega\,\rho_{mk}\delta v_{mk}u_k\frac{\partial v_{mi}}{\partial x_i}\\ &= \int_\Gamma d\Gamma\,\mathbf{u}(\rho_m\mathbf{v}_m\cdot\mathbf{n})\cdot\delta\mathbf{v}_m - \int_\Omega d\Omega\,(\rho_m\mathbf{v}_m\cdot\nabla)\mathbf{u}\cdot\delta\mathbf{v}_m\end{aligned} \tag{4.1.31}$$

and

$$\begin{aligned}\int_\Omega d\Omega\,\mathbf{u}\cdot(\mathbf{v}_m\cdot\nabla)(\delta\alpha\mathbf{v}_m) &= \int_\Omega d\Omega\, u_k v_{mi}\frac{\partial}{\partial x_i}(\delta\alpha_k v_{mk})\\ &= \int_\Omega d\Omega\,\frac{\partial}{\partial x_i}(u_k v_{mi}\delta\alpha_k v_{mk}) - \int_\Omega d\Omega\,\delta\alpha_k v_{mk}\frac{\partial}{\partial x_i}(u_k v_{mi})\\ &= \int_\Gamma d\Gamma\,\mathsf{n}_i u_k v_{mi}\delta\alpha_k v_{mk} - \int_\Omega d\Omega\,\delta\alpha_k v_{mk}v_{mi}\frac{\partial u_k}{\partial x_i}\\ &\quad - \int_\Omega d\Omega\,\delta\alpha_k v_{mk}u_k\frac{\partial v_{mi}}{\partial x_i}\\ &= \int_\Gamma d\Gamma\,\mathbf{u}(\mathbf{v}_m\cdot\mathbf{n})\cdot\mathbf{v}_m\delta\alpha - \int_\Omega d\Omega\,(\mathbf{v}_m\cdot\nabla)\mathbf{u}\cdot\mathbf{v}_m\delta\alpha.\end{aligned} \tag{4.1.32}$$

Applying the tensor-vector identity (Clarke, 2011), the divergence theorem and a property of the colon product, demonstrated below,

$$\begin{aligned}
\nabla\mathbf{u} : \mathsf{D}(\delta\mathbf{v}_m) &= \nabla\mathbf{u} : \frac{1}{2}\left(\nabla\delta\mathbf{v}_m + (\nabla\delta\mathbf{v}_m)^T\right) \\
&= \frac{1}{2}\left(\nabla\mathbf{u} : \nabla\delta\mathbf{v}_m + \nabla\mathbf{u} : (\nabla\delta\mathbf{v}_m)^T\right) \\
&= \frac{1}{2}\left(\nabla\mathbf{u} : \nabla\delta\mathbf{v}_m + (\nabla\mathbf{u})^T : \nabla\delta\mathbf{v}_m\right) \\
&= \frac{1}{2}\left(\nabla\mathbf{u} + (\nabla\mathbf{u})^T\right) : \nabla\delta\mathbf{v}_m \\
&= \mathsf{D}(\mathbf{u}) : \nabla\delta\mathbf{v}_m, \qquad (4.1.33)
\end{aligned}$$

the term containing μ_m can be written as,

$$\begin{aligned}
\int_\Omega d\Omega\, \mathbf{u}\cdot\nabla\cdot\left(2\mu_m\mathsf{D}(\delta\mathbf{v}_m)\right) &= \int_\Omega d\Omega\, \nabla\cdot\left(2\mu_m\mathsf{D}(\delta\mathbf{v}_m)\cdot\mathbf{u}\right) - \int_\Omega d\Omega\, \nabla\mathbf{u} : 2\mu_m\mathsf{D}(\delta\mathbf{v}_m) \\
&= \int_\Gamma d\Gamma\, 2\mu_m\mathbf{n}\cdot\mathsf{D}(\delta\mathbf{v}_m)\cdot\mathbf{u} - \int_\Omega d\Omega\, 2\mu_m\mathsf{D}(\mathbf{u}) : \nabla\delta\mathbf{v}_m \\
&= \int_\Gamma d\Gamma\, 2\mu_m\mathbf{n}\cdot\mathsf{D}(\delta\mathbf{v}_m)\cdot\mathbf{u} - \int_\Omega d\Omega\, \nabla\cdot\left(2\mu_m\mathsf{D}(\mathbf{u})\cdot\delta\mathbf{v}_m\right) \\
&\quad + \int_\Omega d\Omega\, \nabla\cdot\left(2\mu_m\mathsf{D}(\mathbf{u})\right)\cdot\delta\mathbf{v}_m \\
&= \int_\Gamma d\Gamma\, 2\mu_m\mathbf{n}\cdot\mathsf{D}(\delta\mathbf{v}_m)\cdot\mathbf{u} - \int_\Gamma d\Gamma\, 2\mu_m\mathbf{n}\cdot\mathsf{D}(\mathbf{u})\cdot\delta\mathbf{v}_m \\
&\quad + \int_\Omega d\Omega\, \nabla\cdot\left(2\mu_m\mathsf{D}(\mathbf{u})\right)\cdot\delta\mathbf{v}_m. \qquad (4.1.34)
\end{aligned}$$

Applying the product rule and divergence theorem, the remaining terms in Eqn. (4.1.25) can be written as,

$$\begin{aligned}
\int_\Omega d\Omega\, q\nabla\cdot(\rho_m\delta\mathbf{v}_m) &= \int_\Omega d\Omega\, \nabla\cdot(q\rho_m\delta\mathbf{v}_m) - \int_\Omega d\Omega\, \nabla q\cdot\rho_m\delta\mathbf{v}_m \\
&= \int_\Gamma d\Gamma\, q\rho_m\mathbf{n}\cdot\delta\mathbf{v}_m - \int_\Omega d\Omega\, \rho_m\nabla q\cdot\delta\mathbf{v}_m, \qquad (4.1.35)
\end{aligned}$$

$$\begin{aligned}
\int_\Omega d\Omega\, \beta\nabla\cdot(\alpha\delta\mathbf{v}_m) &= \int_\Omega d\Omega\, \nabla\cdot(\beta\alpha\delta\mathbf{v}_m) - \int_\Omega d\Omega\, \nabla\beta\cdot\alpha\delta\mathbf{v}_m \\
&= \int_\Gamma d\Gamma\, \alpha\beta\mathbf{n}\cdot\delta\mathbf{v}_m - \int_\Omega d\Omega\, \alpha\nabla\beta\cdot\delta\mathbf{v}_m, \qquad (4.1.36)
\end{aligned}$$

$$\begin{aligned}
\int_\Omega d\Omega\, \mathbf{u}\cdot\nabla(\rho_m\delta p_m) &= \int_\Omega d\Omega\, \nabla\cdot(\mathbf{u}\rho_m\delta p_m) - \int_\Omega d\Omega\, \nabla\cdot\mathbf{u}\rho_m\delta p_m \\
&= \int_\Gamma d\Gamma\, \rho_m\mathbf{u}\cdot\mathbf{n}\delta p_m - \int_\Omega d\Omega\, \nabla\cdot\rho_m\mathbf{u}\delta p_m, \qquad (4.1.37)
\end{aligned}$$

$$\int_\Omega d\Omega\, \mathbf{u}\cdot\nabla(\delta\alpha p_m) = \int_\Omega d\Omega\, \nabla\cdot(\mathbf{u}\delta\alpha p_m) - \int_\Omega d\Omega\, \nabla\cdot\mathbf{u}\delta\alpha p_m$$
$$= \int_\Gamma d\Gamma\, \mathbf{u}\cdot\mathbf{n}\delta\alpha p_m - \int_\Omega d\Omega\, \nabla\cdot\mathbf{u}\delta\alpha p_m, \tag{4.1.38}$$

$$\int_\Omega d\Omega\, \mathbf{u}\cdot\nabla\cdot\big(\delta\alpha\rho_d\mathbf{v}_{dj}\mathbf{v}_{dj}(2\alpha k_1-1)\big) = \int_\Omega d\Omega\, \nabla\cdot\big(\mathbf{u}\delta\alpha\rho_d\mathbf{v}_{dj}\cdot\mathbf{v}_{dj}(2\alpha k_1-1)\big)$$
$$- \int_\Omega d\Omega\, \nabla\cdot\mathbf{u}\delta\alpha\rho_d\mathbf{v}_{dj}\cdot\mathbf{v}_{dj}(2\alpha k_1-1)$$
$$= \int_\Gamma d\Gamma\, \mathbf{u}\cdot\mathbf{n}\rho_d\mathbf{v}_{dj}\cdot\mathbf{v}_{dj}(2\alpha k_1-1)\delta\alpha$$
$$- \int_\Omega d\Omega\, \nabla\cdot\mathbf{u}\rho_d\mathbf{v}_{dj}\cdot\mathbf{v}_{dj}(2\alpha k_1-1)\delta\alpha, \tag{4.1.39}$$

$$\int_\Omega d\Omega\, q\nabla\cdot(\delta\alpha\mathbf{v}_m) = \int_\Omega d\Omega\, \nabla\cdot(q\delta\alpha\mathbf{v}_m) - \int_\Omega d\Omega\, \nabla q\cdot(\delta\alpha\mathbf{v}_m)$$
$$= \int_\Gamma d\Gamma\, q\mathbf{v}_m\cdot\mathbf{n}\delta\alpha - \int_\Omega d\Omega\, (\mathbf{v}_m\cdot\nabla)q\delta\alpha, \tag{4.1.40}$$

$$\int_\Omega d\Omega\, \beta\nabla\cdot(\delta\alpha\mathbf{v}_m) = \int_\Omega d\Omega\, \nabla\cdot(\beta\delta\alpha\mathbf{v}_m) - \int_\Omega d\Omega\, \nabla\beta\cdot\delta\alpha\mathbf{v}_m$$
$$= \int_\Gamma d\Gamma\, \beta\mathbf{v}_m\cdot\mathbf{n}\delta\alpha - \int_\Omega d\Omega\, (\mathbf{v}_m\cdot\nabla)\beta\delta\alpha, \tag{4.1.41}$$

$$\int_\Omega d\Omega\, \beta\nabla\cdot\big(\delta\alpha\mathbf{v}_{dj}(\alpha k_1-1)\big) = \int_\Omega d\Omega\, \nabla\cdot\big(\beta\delta\alpha\mathbf{v}_{dj}(\alpha k_1-1)\big)$$
$$- \int_\Omega d\Omega\, \nabla\beta\cdot\delta\alpha\mathbf{v}_{dj}(\alpha k_1-1)$$
$$= \int_\Gamma d\Gamma\, \beta(\alpha k_1-1)\mathbf{v}_{dj}\cdot\mathbf{n}\delta\alpha$$
$$- \int_\Omega d\Omega\, \delta\alpha(\alpha k_1-1)(\mathbf{v}_{dj}\cdot\nabla)\beta \tag{4.1.42}$$

and

$$\int_\Omega d\Omega\, \beta\nabla\cdot K\nabla\delta\alpha = \int_\Omega d\Omega\, \nabla\cdot(\beta K\nabla\delta\alpha) - \int_\Omega d\Omega\, \nabla\beta\cdot K\nabla\delta\alpha$$
$$= \int_\Gamma d\Gamma\, \beta K\mathbf{n}\cdot\nabla\delta\alpha - \int_\Omega d\Omega\, \nabla\cdot(K\delta\alpha\nabla\beta) + \int_\Omega d\Omega\, \delta\alpha\nabla\cdot K\nabla\beta$$
$$= \int_\Gamma d\Gamma\, \beta K(\mathbf{n}\cdot\nabla)\delta\alpha - \int_\Gamma d\Gamma\, K\delta\alpha(\mathbf{n}\cdot\nabla)\beta + \int_\Omega d\Omega\, \delta\alpha\nabla\cdot K\nabla\beta. \tag{4.1.43}$$

Eqn. (4.1.25) can now be reformulated as,

$$\begin{aligned}
&\int_\Gamma d\Gamma\,\frac{\partial J_\Gamma}{\partial \mathbf{v}_m}\cdot\delta\mathbf{v}_m+\int_\Omega d\Omega\,\frac{\partial J_\Omega}{\partial \mathbf{v}_m}\cdot\delta\mathbf{v}_m+\int_\Gamma d\Gamma\,\frac{\partial J_\Gamma}{\partial p_m}\delta p_m+\int_\Omega d\Omega\,\frac{\partial J_\Omega}{\partial p_m}\delta p_m\\
&+\int_\Gamma d\Gamma\,\frac{\partial J_\Gamma}{\partial \alpha}\delta\alpha+\int_\Omega d\Omega\,\frac{\partial J_\Omega}{\partial \alpha}\delta\alpha+\int_\Gamma d\Gamma\,\mathbf{n}(\mathbf{u}\cdot\rho_m\mathbf{v}_m)\cdot\delta\mathbf{v}_m-\int_\Omega d\Omega\,\nabla\mathbf{u}\cdot(\rho_m\mathbf{v}_m)\cdot\delta\mathbf{v}_m\\
&+\int_\Gamma d\Gamma\,\mathbf{u}(\rho_m\mathbf{v}_m\cdot\mathbf{n})\cdot\delta\mathbf{v}_m-\int_\Omega d\Omega\,(\rho_m\mathbf{v}_m\cdot\nabla)\mathbf{u}\cdot\delta\mathbf{v}_m-\int_\Gamma d\Gamma\,2\mu_m\mathbf{n}\cdot\mathsf{D}(\delta\mathbf{v}_m)\cdot\mathbf{u}\\
&+\int_\Gamma d\Gamma\,2\mu_m\mathbf{n}\cdot\mathsf{D}(\mathbf{u})\cdot\delta\mathbf{v}_m-\int_\Omega d\Omega\,\nabla\cdot\big(2\mu_m\mathsf{D}(\mathbf{u})\big)\cdot\delta\mathbf{v}_m+\int_\Omega d\Omega\,\aleph\rho_m\mathbf{u}\cdot\delta\mathbf{v}_m\\
&-\int_\Gamma d\Gamma\,q\rho_m\mathbf{n}\cdot\delta\mathbf{v}_m+\int_\Omega d\Omega\,\rho_m\nabla q\cdot\delta\mathbf{v}_m+\int_\Gamma d\Gamma\,\alpha\beta\mathbf{n}\cdot\delta\mathbf{v}_m-\int_\Omega d\Omega\,\alpha\nabla\beta\cdot\delta\mathbf{v}_m\\
&+\int_\Gamma d\Gamma\,\rho_m\mathbf{u}\cdot\mathbf{n}\delta p_m-\int_\Omega d\Omega\,\nabla\cdot\rho_m\mathbf{u}\delta p_m\\
&-\int_\Gamma d\Gamma\,\mathbf{u}\cdot\mathbf{n}\rho_d\mathbf{v}_{dj}\cdot\mathbf{v}_{dj}(2\alpha k_1-1)\delta\alpha+\int_\Omega d\Omega\,\nabla\cdot\mathbf{u}\rho_d\mathbf{v}_{dj}\cdot\mathbf{v}_{dj}(2\alpha k_1-1)\delta\alpha\\
&+(\rho_d-\rho_c)\int_\Gamma d\Gamma\,\mathbf{u}(\mathbf{v}_m\cdot\mathbf{n})\cdot\mathbf{v}_m\delta\alpha-(\rho_d-\rho_c)\int_\Omega d\Omega\,(\mathbf{v}_m\cdot\nabla)\mathbf{u}\cdot\mathbf{v}_m\delta\alpha\\
&+(\rho_d-\rho_c)\int_\Gamma d\Gamma\,p_m\mathbf{u}\cdot\mathbf{n}\delta\alpha-(\rho_d-\rho_c)\int_\Omega d\Omega\,p_m\nabla\cdot\mathbf{u}\delta\alpha+(\rho_d-\rho_c)\int_\Omega d\Omega\,\mathbf{u}\cdot(\aleph\mathbf{v}_m-\mathbf{g})\delta\alpha\\
&-(\rho_d-\rho_c)\int_\Gamma d\Gamma\,q\mathbf{v}_m\cdot\mathbf{n}\delta\alpha+(\rho_d-\rho_c)\int_\Omega d\Omega\,(\mathbf{v}_m\cdot\nabla)q\delta\alpha+\int_\Gamma d\Gamma\,\beta\mathbf{v}_m\cdot\mathbf{n}\delta\alpha\\
&-\int_\Omega d\Omega\,(\mathbf{v}_m\cdot\nabla)\beta\delta\alpha-\int_\Gamma d\Gamma\,\beta(\alpha k_1-1)\mathbf{v}_{dj}\cdot\mathbf{n}\delta\alpha+\int_\Omega d\Omega\,\delta\alpha(\alpha k_1-1)(\mathbf{v}_{dj}\cdot\nabla)\beta\\
&-\int_\Gamma d\Gamma\,\beta K(\mathbf{n}\cdot\nabla)\delta\alpha+\int_\Gamma d\Gamma\,K\delta\alpha(\mathbf{n}\cdot\nabla)\beta-\int_\Omega d\Omega\,\delta\alpha\nabla\cdot K\nabla\beta=0,
\end{aligned}\tag{4.1.44}$$

which is rearranged as,

$$\begin{aligned}
&\int_\Gamma d\Gamma\left(\mathbf{n}(\mathbf{u}\cdot\rho_m\mathbf{v}_m)+\mathbf{u}(\rho_m\mathbf{v}_m\cdot\mathbf{n})+2\mu_m\mathbf{n}\cdot\mathsf{D}(\mathbf{u})-q\rho_m\mathbf{n}+\alpha\beta\mathbf{n}+\frac{\partial J_\Gamma}{\partial\mathbf{v}_m}\right)\cdot\delta\mathbf{v}_m\\
&-\int_\Gamma d\Gamma 2\mu_m\mathbf{n}\cdot\mathsf{D}(\delta\mathbf{v}_m)\cdot\mathbf{u}+\int_\Omega d\Omega\Big(-\nabla\mathbf{u}\cdot(\rho_m\mathbf{v}_m)\\
&-(\rho_m\mathbf{v}_m\cdot\nabla)\mathbf{u}-\nabla\cdot\big(2\mu_m\mathsf{D}(\mathbf{u})\big)+\aleph\rho_m\mathbf{u}+\rho_m\nabla q-\alpha\nabla\beta+\frac{\partial J_\Omega}{\partial\mathbf{v}_m}\Big)\cdot\delta\mathbf{v}_m\\
&+\int_\Gamma d\Gamma\left(\rho_m\mathbf{u}\cdot\mathbf{n}+\frac{\partial J_\Gamma}{\partial p_m}\right)\delta p_m+\int_\Omega d\Omega\left(-\nabla\cdot\rho_m\mathbf{u}+\frac{\partial J_\Omega}{\partial p_m}\right)\delta p_m\\
&+\int_\Gamma d\Gamma\Big((\rho_d-\rho_c)\big(\mathbf{u}(\mathbf{v}_m\cdot\mathbf{n})\cdot\mathbf{v}_m+p_m\mathbf{u}\cdot\mathbf{n}-q\mathbf{v}_m\cdot\mathbf{n}\big)-\mathbf{u}\cdot\mathbf{n}\rho_d\mathbf{v}_{dj}\cdot\mathbf{v}_{dj}(2\alpha k_1-1)\\
&+\beta\mathbf{v}_m\cdot\mathbf{n}-\beta(\alpha k_1-1)\mathbf{v}_{dj}\cdot\mathbf{n}+K(\mathbf{n}\cdot\nabla)\beta+\frac{\partial J_\Gamma}{\partial\alpha}\Big)\delta\alpha-\int_\Gamma d\Gamma\beta K(\mathbf{n}\cdot\nabla)\delta\alpha\\
&+\int_\Omega d\Omega\Big((\rho_d-\rho_c)\big(-(\mathbf{v}_m\cdot\nabla)\mathbf{u}\cdot\mathbf{v}_m-p_m\nabla\cdot\mathbf{u}+\mathbf{u}\cdot(\aleph\mathbf{v}_m-\mathbf{g})+(\mathbf{v}_m\cdot\nabla)q\big)\\
&+\rho_d\mathbf{v}_{dj}\cdot\mathbf{v}_{dj}(2\alpha k_1-1)\nabla\cdot\mathbf{u}-(\mathbf{v}_m\cdot\nabla)\beta+(\alpha k_1-1)(\mathbf{v}_{dj}\cdot\nabla)\beta-\nabla\cdot K\nabla\beta+\frac{\partial J_\Omega}{\partial\alpha}\Big)\delta\alpha=0.
\end{aligned}\tag{4.1.45}$$

In order to satisfy Eqn. (4.1.45) in general, the integrals must vanish individually. The adjoint drift flux equations are deduced from the integrals over the interior of the domain:

$$-(\rho_m \mathbf{v}_m \cdot \nabla)\mathbf{u} = -\rho_m \nabla q + \nabla \mathbf{u} \cdot (\rho_m \mathbf{v}_m) + \nabla \cdot \big(2\mu_m \mathsf{D}(\mathbf{u})\big) + \alpha \nabla \beta - \aleph \rho_m \mathbf{u} - \frac{\partial J_\Omega}{\partial \mathbf{v}_m}, \tag{4.1.46a}$$

$$\nabla \cdot (\rho_m \mathbf{u}) = \frac{\partial J_\Omega}{\partial p_m}, \tag{4.1.46b}$$

$$-(\mathbf{v}_m \cdot \nabla)\beta = -(\alpha k_1 - 1)(\mathbf{v}_{dj} \cdot \nabla)\beta + \nabla \cdot K \nabla \beta + S_1 + S_2 - \frac{\partial J_\Omega}{\partial \alpha}, \tag{4.1.46c}$$

where:

$$S_1 = (\rho_d - \rho_c)\big((\mathbf{v}_m \cdot \nabla)(\mathbf{u} \cdot \mathbf{v}_m - q) - \mathbf{u} \cdot (\aleph \mathbf{v}_m - \mathbf{g})\big), \tag{4.1.47a}$$

$$S_2 = \big((\rho_d - \rho_c)p_m - (2\alpha k_1 - 1)\rho_d \mathbf{v}_{dj} \cdot \mathbf{v}_{dj}\big)\nabla \cdot \mathbf{u}, \tag{4.1.47b}$$

and the boundary conditions for the adjoint variables are deduced from the surface integrals:

$$\int_\Gamma d\Gamma \left(\mathbf{n}(\mathbf{u} \cdot \rho_m \mathbf{v}_m) + \rho_m v_{m\mathsf{n}} \mathbf{u} + 2\mu_m \mathbf{n} \cdot \mathsf{D}(\mathbf{u}) - q\rho_m \mathbf{n} + \alpha\beta \mathbf{n} + \frac{\partial J_\Gamma}{\partial \mathbf{v}_m}\right) \cdot \delta \mathbf{v}_m - \int_\Gamma d\Gamma 2\mu_m \mathbf{n} \cdot \mathsf{D}(\delta \mathbf{v}_m) \cdot \mathbf{u} = 0, \tag{4.1.48a}$$

$$\int_\Gamma d\Gamma \left(\rho_m u_\mathsf{n} + \frac{\partial J_\Gamma}{\partial p_m}\right) \delta p_m = 0, \tag{4.1.48b}$$

$$\int_\Gamma d\Gamma \bigg(\big(v_{m\mathsf{n}} - (\alpha k_1 - 1)v_{dj\mathsf{n}}\big)\beta + K(\mathbf{n} \cdot \nabla)\beta + C_1 u_\mathsf{n} + C_2 v_{m\mathsf{n}} + \frac{\partial J_\Gamma}{\partial \alpha}\bigg)\delta\alpha - \int_\Gamma d\Gamma \beta K(\mathbf{n} \cdot \nabla)\delta\alpha = 0, \tag{4.1.48c}$$

where:

$$C_1 = \big((\rho_d - \rho_c)p_m - (2\alpha k_1 - 1)\rho_d \mathbf{v}_{dj} \cdot \mathbf{v}_{dj}\big), \tag{4.1.49a}$$

$$C_2 = (\rho_d - \rho_c)(\mathbf{u} \cdot \mathbf{v}_m - q), \tag{4.1.49b}$$

and $u_\mathsf{n} = \mathbf{u} \cdot \mathbf{n}$, $v_{m\mathsf{n}} = \mathbf{v}_m \cdot \mathbf{n}$ and $v_{dj\mathsf{n}} = \mathbf{v}_{dj} \cdot \mathbf{n}$ are the normal components of the adjoint, primal and drift velocities, respectively. This is the general form of the adjoint equation system for the steady state drift flux equations with Darcy porosity term and frozen viscosity.

4.1.2 Optimising the Design

The topological sensitivity of the Lagrange function at cell i is computed from Eqn. (4.1.6) as,

$$\begin{aligned}\frac{\partial \mathscr{L}}{\partial \aleph_i} &= \frac{\partial J}{\partial \aleph_i} + \int_\Omega (\mathbf{u}, q, \beta) \frac{\partial \mathcal{R}}{\partial \aleph_i}\, d\Omega \\ &= 0 + \int_\Omega (\mathbf{u}, q, \beta)(\rho_m \mathbf{v}_m, 0, 0)^T V_i\, d\Omega \\ &= \rho_{mi}\, \mathbf{u}_i \cdot \mathbf{v}_{mi} V_i, \end{aligned} \tag{4.1.50}$$

where V_i is the volume of cell i, and porosity is calculated using a steepest ascent algorithm,

$$\begin{aligned}\aleph_i &= \frac{\partial \mathscr{L}}{\partial \aleph_i}\delta, \\ &= \rho_{mi}\, \mathbf{u}_i \cdot \mathbf{v}_{mi} V_i \delta, \end{aligned} \tag{4.1.51}$$

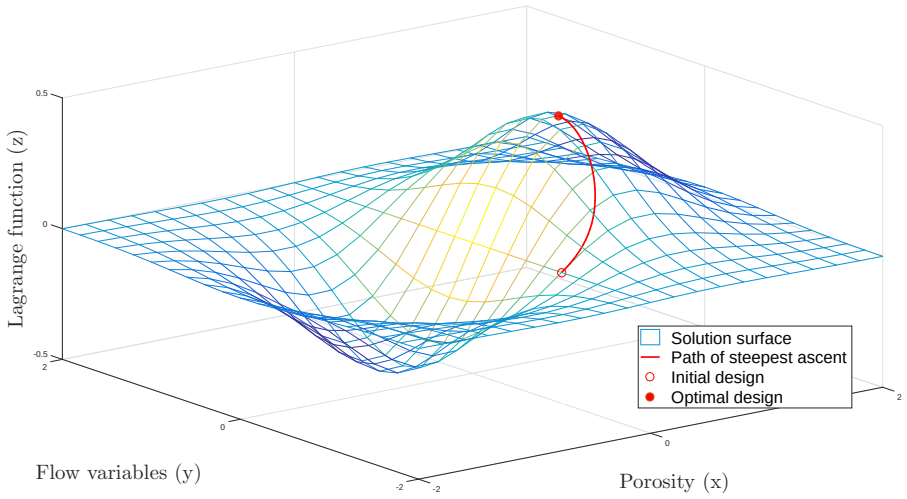
where δ is the step length. In order to find the optimal design, the following steps are repeated for a range of critical porosity values, $\aleph_{crit}$:

1. Convert cells with $\aleph \geqslant \aleph_{crit}$ into walls to create a new geometry.
2. Solve the drift flux equations on the new geometry to a quasi-steady state.
3. Calculate the objective function.

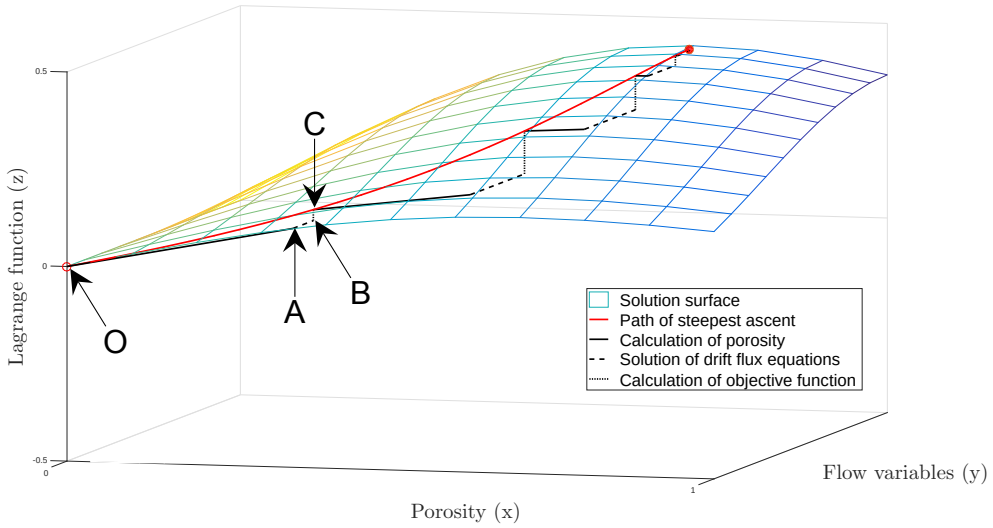
The optimisation procedure is illustrated with the aid of Figures 4.1a and 4.1b. The former is a graphical representation of the Lagrange function solution space as a 3-dimensional surface, where the notional surface topography and axes values are indicative for discussion purposes. In this figure, the red line represents the path of steepest ascent along the solution space from the empty-red circle at the lower end, representing an initial HDVS design, to the filled-red circle at the upper end, representing an optimal design. Figure 4.1b is a sub-set of the solution space in Figure 4.1a, between x = [0,1] and y = [-1,0], rotated clockwise through 52.5°. The optimisation procedure is described, using Figure 4.1b in conjunction with the flow chart in Figure 4.2, as follows.

This page is intentionally left blank, to show the procedure on facing pages.

The optimisation procedure goes from O $\rightarrow$ A $\rightarrow$ B $\rightarrow$ C, and can be repeated from C to the filled-red circle but, in this methodology, stops at C. It consists of a loop, Figure 4.2, from A $\rightarrow$ a point on AB $\rightarrow$ the point on the solution surface directly above it, with the path to the optimal design being from A $\rightarrow$ B $\rightarrow$ C.



(a)



(b)

Figure 4.1: Graphical representation of (a) the Lagrange function solution space and (b) the optimisation procedure.

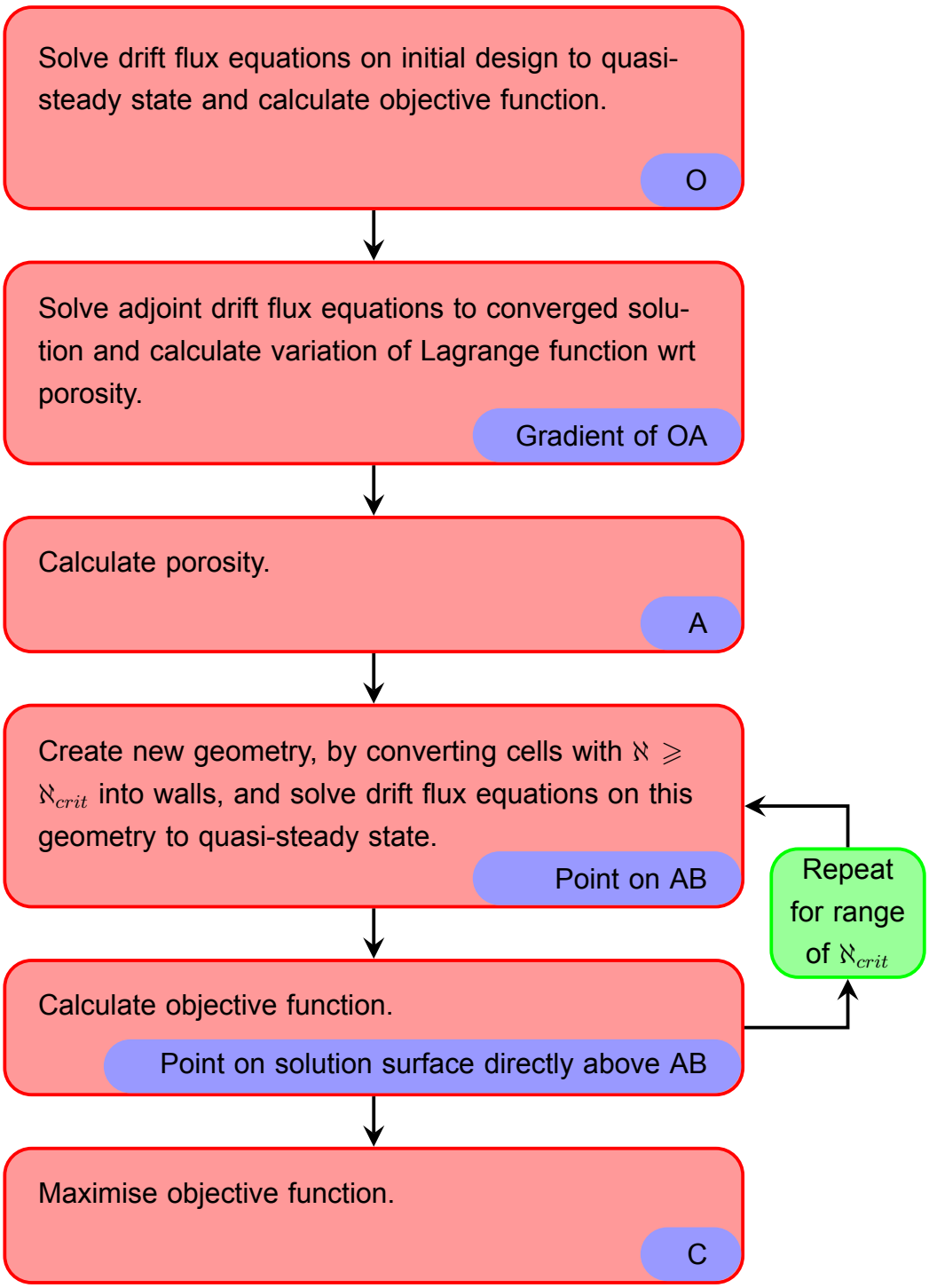


Figure 4.2: Flow chart of optimisation procedure. State of system represented by examples in blue boxes, corresponding to symbols in Figure 4.1b.

4.2 Ducted Flows

In the scope of this research, there is no contribution to the objective function from the interior of the domain. So, in the cases where the objective function only involves integrals over the surface of the flow domain rather than over its interior, the adjoint equations reduce to:

$$-(\rho_m \mathbf{v}_m \cdot \nabla)\mathbf{u} = -\rho_m \nabla q + \nabla \mathbf{u} \cdot (\rho_m \mathbf{v}_m) + \nabla \cdot \big(2\mu_m \mathsf{D}(\mathbf{u})\big) + \alpha \nabla \beta - \aleph \rho_m \mathbf{u}, \tag{4.2.1a}$$

$$\nabla \cdot (\rho_m \mathbf{u}) = 0, \tag{4.2.1b}$$

$$-(\mathbf{v}_m \cdot \nabla)\beta = -(\alpha k_1 - 1)(\mathbf{v}_{dj} \cdot \nabla)\beta + \nabla \cdot K \nabla \beta + S_1 + S_2. \tag{4.2.1c}$$

These equations no longer depend on the objective function, so when switching from one optimisation objective to another, the solver remains unchanged and only the boundary conditions have to be adapted to the specific objective function. For the adjoint boundary conditions, the terms in Eqn. (4.1.48a) involving the strain rate tensor, D, can be rewritten as,

$$\begin{aligned}\int_\Gamma d\Gamma\, 2\mu_m \mathbf{n} \cdot \big(\mathsf{D}(\mathbf{u}) \cdot \delta\mathbf{v}_m - \mathsf{D}(\delta\mathbf{v}_m) \cdot \mathbf{u}\big) &= \int_\Gamma d\Gamma\, \mu_m \mathbf{n} \cdot \Big(\big(\nabla \mathbf{u} + (\nabla \mathbf{u})^T\big) \cdot \delta\mathbf{v}_m \\ &\qquad - \big(\nabla \delta\mathbf{v}_m + (\nabla \delta\mathbf{v}_m)^T\big) \cdot \mathbf{u}\Big) \\ &= \int_\Gamma d\Gamma \mu_m \mathbf{n} \cdot \big((\nabla \mathbf{u})^T \cdot \delta\mathbf{v}_m - (\nabla \delta\mathbf{v}_m)^T \cdot \mathbf{u}\big) \\ &\qquad + \int_\Gamma d\Gamma\, \mu_m \mathbf{n} \cdot (\nabla \mathbf{u} \cdot \delta\mathbf{v}_m - \nabla \delta\mathbf{v}_m \cdot \mathbf{u}).\end{aligned} \tag{4.2.2}$$

Applying the product rule, divergence theorem and $\delta\mathbf{v}_m = 0$ along the boundary edges, $\partial\Gamma$, and using the Einstein notation for clarity, the terms in the first integral

on the RHS can be written as,

$$\begin{aligned}
\int_\Gamma d\Gamma\, \mu_m \mathbf{n}\cdot(\nabla\mathbf{u})^T\cdot\delta\mathbf{v}_m &= \int_\Gamma d\Gamma\, \mu_m \mathsf{n}_i(\partial_k u_i)\delta v_{mk} \\
&= \int_\Gamma d\Gamma\, \partial_k(\mu_m \mathsf{n}_i u_i \delta v_{mk}) - \int_\Gamma d\Gamma\, u_i\partial_k(\mu_m \mathsf{n}_i \delta v_{mk}) \\
&= \int_{\partial\Gamma} \partial\Gamma\, \tilde{\mathsf{n}}_k \mu_m \mathsf{n}_i u_i \delta v_{mk} - \int_\Gamma d\Gamma\, u_i \mathsf{n}_i \delta v_{mk}\partial_k\mu_m \\
&\quad - \int_\Gamma d\Gamma\, u_i\mu_m\partial_k(\mathsf{n}_i\delta v_{mk}) \\
&= 0 - \int_\Gamma d\Gamma\, \partial_k\mu_m u_i \mathsf{n}_i \delta v_{mk} - \int_\Gamma d\Gamma\, u_i\mu_m\mathsf{n}_i\partial_k\delta v_{mk} \\
&\quad - \int_\Gamma d\Gamma\, u_i\mu_m\delta v_{mk}\partial_k\mathsf{n}_i \\
&= -\int_\Gamma d\Gamma\, \nabla\mu_m\cdot u_\mathsf{n}\delta\mathbf{v}_m - \int_\Gamma d\Gamma\, \mu_m u_\mathsf{n}\nabla\cdot\delta\mathbf{v}_m \\
&\quad - \int_\Gamma d\Gamma\, \mu_m(\mathbf{u}\cdot\nabla\mathbf{n})\cdot\delta\mathbf{v}_m
\end{aligned} \tag{4.2.3}$$

and

$$\begin{aligned}
\int_\Gamma d\Gamma\, \mu_m \mathbf{n}\cdot(\nabla\delta\mathbf{v}_m)^T\cdot\mathbf{u} &= \int_\Gamma d\Gamma\, \mu_m \mathsf{n}_i(\partial_k \delta v_{mi})u_k \\
&= \int_\Gamma d\Gamma\, \partial_k(\mu_m \mathsf{n}_i \delta v_{mi} u_k) - \int_\Gamma d\Gamma\, \delta v_{mi}\partial_k(\mu_m \mathsf{n}_i u_k) \\
&= \int_{\partial\Gamma} \partial\Gamma\, \tilde{\mathsf{n}}_k \mu_m \mathsf{n}_i \delta v_{mi} u_k - \int_\Gamma d\Gamma\, \delta v_{mi} \mathsf{n}_i u_k \partial_k\mu_m \\
&\quad - \int_\Gamma d\Gamma\, \delta v_{mi}\mu_m\partial_k(\mathsf{n}_i u_k) \\
&= 0 - \int_\Gamma d\Gamma\, \partial_k\mu_m \delta v_{mi} \mathsf{n}_i u_k - \int_\Gamma d\Gamma\, \delta v_{mi}\mu_m\mathsf{n}_i\partial_k u_k \\
&\quad - \int_\Gamma d\Gamma\, \delta v_{mi}\mu_m u_k\partial_k\mathsf{n}_i \\
&= -\int_\Gamma d\Gamma\, \nabla\mu_m\cdot\delta v_\mathsf{n}\mathbf{u} - \int_\Gamma d\Gamma\, \mu_m\delta v_\mathsf{n}\nabla\cdot\mathbf{u} \\
&\quad - \int_\Gamma d\Gamma\, \mu_m(\delta\mathbf{v}_m\cdot\nabla\mathbf{n})\cdot\mathbf{u},
\end{aligned} \tag{4.2.4}$$

respectively, where $\tilde{\mathsf{n}}_k$ is the k^{th} component of a vector normal to the boundary edge. As $\delta\mathbf{v}_m$ and $\mathbf{u}$ are divergence-less fields and $\mathbf{n}$ is a constant vector, $\nabla\cdot\delta\mathbf{v}_m = 0$, $\nabla\cdot\mathbf{u} = 0$ and $\nabla\mathbf{n} = 0$. Hence, the second and third terms in Eqns. (4.2.3) and (4.2.4) cancel and Eqn (4.2.2) reduces to,

$$\begin{aligned}
\int_\Gamma d\Gamma\, 2\mu_m\mathbf{n}\cdot\big(\mathsf{D}(\mathbf{u})\cdot\delta\mathbf{v}_m - \mathsf{D}(\delta\mathbf{v}_m)\cdot\mathbf{u}\big) &= \int_\Gamma d\Gamma\, \mu_m\big((\mathbf{n}\cdot\nabla)\mathbf{u}\cdot\delta\mathbf{v}_m - (\mathbf{n}\cdot\nabla)\delta\mathbf{v}_m\cdot\mathbf{u}\big) \\
&\quad - \int_\Gamma d\Gamma\, \nabla\mu_m\cdot(u_\mathsf{n}\delta\mathbf{v}_m - \delta v_{m\mathsf{n}}\mathbf{u}).
\end{aligned} \tag{4.2.5}$$

The second integral on the RHS of this equation can be neglected, because the product of $\nabla\mu_m$ with the difference term in brackets is assumed to be of second order (Othmer, 2008) and therefore the adjoint boundary conditions, Eqns. (4.1.48), reduce to:

$$\int_\Gamma d\Gamma \left(\mathbf{n}(\mathbf{u}\cdot\rho_m\mathbf{v}_m) + \rho_m v_{m\mathsf{n}}\mathbf{u} + \mu_m(\mathbf{n}\cdot\nabla)\mathbf{u} - q\rho_m\mathbf{n} + \alpha\beta\mathbf{n} + \frac{\partial J_\Gamma}{\partial \mathbf{v}_m}\right)\cdot\delta\mathbf{v}_m - \int_\Gamma d\Gamma\mu_m(\mathbf{n}\cdot\nabla)\delta\mathbf{v}_m\cdot\mathbf{u} = 0, \tag{4.2.6a}$$

$$\int_\Gamma d\Gamma \left(\rho_m u_{\mathsf{n}} + \frac{\partial J_\Gamma}{\partial p_m}\right)\delta p_m = 0, \tag{4.2.6b}$$

$$\int_\Gamma d\Gamma \left(\big(v_{m\mathsf{n}} - (\alpha k_1 - 1)v_{dj\mathsf{n}}\big)\beta + K(\mathbf{n}\cdot\nabla)\beta + C_1 u_{\mathsf{n}} + C_2 v_{m\mathsf{n}} + \frac{\partial J_\Gamma}{\partial\alpha}\right)\delta\alpha - \int_\Gamma d\Gamma\beta K(\mathbf{n}\cdot\nabla)\delta\alpha = 0. \tag{4.2.6c}$$

In order to determine the boundary conditions of the adjoint variables, the boundary conditions imposed on the primal variables are listed in Table 4.1.

Table 4.1: Primal boundary conditions.

	$\mathbf{v}_m$	α	p_m
Inlet	fixed value	fixed value	zero gradient
Wall	zero	zero gradient	zero gradient
Underflow	fixed value	zero gradient	zero gradient
Overflow	zero gradient	zero gradient	zero

4.2.1 Adjoint Boundary Conditions at the Inlet

At the inlet, the primal velocity and dispersed-phase volume fraction are usually fixed, so,

$$\delta \mathbf{v}_m = 0 \text{ and } \delta\alpha = 0. \tag{4.2.7}$$

The first integrals in Eqns. (4.2.6a) and (4.2.6c) therefore go to zero and Eqns. (4.2.6) reduce to:

$$\int_\Gamma d\Gamma \mu_m (\mathbf{n}\cdot\nabla)\delta\mathbf{v}_m \cdot \mathbf{u} = 0, \tag{4.2.8a}$$

$$\int_\Gamma d\Gamma \left(\rho_m u_\mathsf{n} + \frac{\partial J_\Gamma}{\partial p_m}\right)\delta p_m = 0, \tag{4.2.8b}$$

$$\int_\Gamma d\Gamma \beta K (\mathbf{n}\cdot\nabla)\delta\alpha = 0. \tag{4.2.8c}$$

Ubbink (1997) demonstrated that, when both fluids are incompressible, $\nabla\cdot\mathbf{v}_m = 0$, and Othmer (2008) demonstrated that, as $\delta\mathbf{v}_{m\mathsf{t}} = 0$ along the inlet, $(\mathbf{n}\cdot\nabla)\delta\mathbf{v}_m = (\mathbf{n}\cdot\nabla)\delta\mathbf{v}_{m\mathsf{t}}$, where $\mathbf{v}_{m\mathsf{t}}$ is the tangential component of the mixture velocity. Hence, Eqns. (4.2.8) reduce to:

$$\int_\Gamma d\Gamma \mu_m (\mathbf{n}\cdot\nabla)\delta\mathbf{v}_{m\mathsf{t}} \cdot \mathbf{u}_\mathsf{t} = 0, \tag{4.2.9a}$$

$$\int_\Gamma d\Gamma \left(\rho_m u_\mathsf{n} + \frac{\partial J_\Gamma}{\partial p_m}\right)\delta p_m = 0, \tag{4.2.9b}$$

$$\int_\Gamma d\Gamma \beta K (\mathbf{n}\cdot\nabla)\delta\alpha = 0, \tag{4.2.9c}$$

where $\mathbf{u}_\mathsf{t}$ is the tangential component of the adjoint velocity, from which we deduce the boundary conditions for the adjoint variables at the inlet to be:

$$\mathbf{u}_\mathsf{t} = 0, \tag{4.2.10a}$$

$$u_\mathsf{n} = -\frac{1}{\rho_m}\frac{\partial J_\Gamma}{\partial p_m}, \tag{4.2.10b}$$

$$\beta = 0. \tag{4.2.10c}$$

Note that these derivations do not impose a condition for q. Since q enters the adjoint drift flux equations in a manner similar to the way p_m enters the primal drift flux equations, the zero gradient boundary condition of p_m at the inlet is applied to q as well,

$$(\mathbf{n}\cdot\nabla)q = 0. \tag{4.2.11}$$

4.2.2 Adjoint Boundary Conditions at the Wall

At the wall, typical primal conditions are zero velocity and zero gradient of the dispersed-phase volume fraction. Therefore, we have,

$$\delta\mathbf{v}_m = 0,\ v_{m\mathsf{n}} = 0 \text{ and } (\mathbf{n}\cdot\nabla)\delta\alpha = 0. \tag{4.2.12}$$

The first integral in Eqn. (4.2.6a) and the second integral in Eqn. (4.2.6c) therefore go to zero and the terms in the first integral in Eqn. (4.2.6c), containing $v_{m\mathsf{n}}$, go to zero. Eqns. (4.2.6) therefore reduce to:

$$\int_\Gamma d\Gamma \mu_m(\mathbf{n}\cdot\nabla)\delta\mathbf{v}_m\cdot\mathbf{u} = 0, \tag{4.2.13a}$$

$$\int_\Gamma d\Gamma \left(\rho_m u_{\mathsf{n}} + \frac{\partial J_\Gamma}{\partial p_m}\right)\delta p_m = 0, \tag{4.2.13b}$$

$$\int_\Gamma d\Gamma \bigg(-(\alpha k_1 - 1)v_{dj\mathsf{n}}\beta + K(\mathbf{n}\cdot\nabla)\beta + C_1 u_{\mathsf{n}} + \frac{\partial J_\Gamma}{\partial\alpha}\bigg)\delta\alpha = 0. \tag{4.2.13c}$$

As at the inlet, the primal velocity does not diverge and $\delta\mathbf{v}_{m\mathsf{t}} = 0$ along the wall, so Eqns. (4.2.13) reduce to:

$$\int_\Gamma d\Gamma \mu_m(\mathbf{n}\cdot\nabla)\delta\mathbf{v}_{m\mathsf{t}}\cdot\mathbf{u}_{\mathsf{t}} = 0, \tag{4.2.14a}$$

$$\int_\Gamma d\Gamma \left(\rho_m u_{\mathsf{n}} + \frac{\partial J_\Gamma}{\partial p_m}\right)\delta p_m = 0, \tag{4.2.14b}$$

$$\int_\Gamma d\Gamma \bigg(-(\alpha k_1 - 1)v_{dj\mathsf{n}}\beta + K(\mathbf{n}\cdot\nabla)\beta + C_1 u_{\mathsf{n}} + \frac{\partial J_\Gamma}{\partial\alpha}\bigg)\delta\alpha = 0, \tag{4.2.14c}$$

from which we deduce the boundary conditions for the adjoint variables at the wall to be:

$$\mathbf{u}_{\mathsf{t}} = 0, \tag{4.2.15a}$$

$$u_{\mathsf{n}} = -\frac{1}{\rho_m}\frac{\partial J_\Gamma}{\partial p_m}, \tag{4.2.15b}$$

$$-(\alpha k_1 - 1)v_{dj\mathsf{n}}\beta + K(\mathbf{n}\cdot\nabla)\beta = -C_1 u_{\mathsf{n}} - \frac{\partial J_\Gamma}{\partial\alpha}. \tag{4.2.15c}$$

Eqn. (4.2.15c) is used to determine β and, as at the inlet, Eqn. (4.2.11) applies.

4.2.3 Adjoint Boundary Conditions at the Underflow

At the underflow, typical primal conditions are fixed velocity and zero gradient of the dispersed-phase volume fraction. Therefore, we have,

$$\delta\mathbf{v}_m = 0 \text{ and } (\mathbf{n}\cdot\nabla)\delta\alpha = 0. \tag{4.2.16}$$

The first integral in Eqn. (4.2.6a) and the second integral in Eqn. (4.2.6c) therefore go to zero and Eqns. (4.2.6) reduce to:

$$\int_\Gamma d\Gamma \mu_m (\mathbf{n}\cdot\nabla)\delta\mathbf{v}_m \cdot \mathbf{u} = 0, \tag{4.2.17a}$$

$$\int_\Gamma d\Gamma \left(\rho_m u_\mathsf{n} + \frac{\partial J_\Gamma}{\partial p_m}\right)\delta p_m = 0, \tag{4.2.17b}$$

$$\int_\Gamma d\Gamma \Big(\big(v_{m\mathsf{n}} - (\alpha k_1 - 1)v_{dj\mathsf{n}}\big)\beta + K(\mathbf{n}\cdot\nabla)\beta + C_1 u_\mathsf{n} + C_2 v_{m\mathsf{n}} + \frac{\partial J_\Gamma}{\partial \alpha}\Big)\delta\alpha = 0. \tag{4.2.17c}$$

As at the inlet, the primal velocity does not diverge and $\delta\mathbf{v}_{m\mathsf{t}} = 0$ along the underflow, so Eqns. (4.2.17) reduce to:

$$\int_\Gamma d\Gamma \mu_m (\mathbf{n}\cdot\nabla)\delta\mathbf{v}_{m\mathsf{t}} \cdot \mathbf{u}_\mathsf{t} = 0, \tag{4.2.18a}$$

$$\int_\Gamma d\Gamma \left(\rho_m u_\mathsf{n} + \frac{\partial J_\Gamma}{\partial p_m}\right)\delta p_m = 0, \tag{4.2.18b}$$

$$\int_\Gamma d\Gamma \Big(\big(v_{m\mathsf{n}} - (\alpha k_1 - 1)v_{dj\mathsf{n}}\big)\beta + K(\mathbf{n}\cdot\nabla)\beta + C_1 u_\mathsf{n} + C_2 v_{m\mathsf{n}} + \frac{\partial J_\Gamma}{\partial \alpha}\Big)\delta\alpha = 0, \tag{4.2.18c}$$

from which we deduce the boundary conditions for the adjoint variables at the underflow to be:

$$\mathbf{u}_\mathsf{t} = 0, \tag{4.2.19a}$$

$$u_\mathsf{n} = -\frac{1}{\rho_m}\frac{\partial J_\Gamma}{\partial p_m}, \tag{4.2.19b}$$

$$\big(v_{m\mathsf{n}} - (\alpha k_1 - 1)v_{dj\mathsf{n}}\big)\beta + K(\mathbf{n}\cdot\nabla)\beta = -C_1 u_\mathsf{n} - C_2 v_{m\mathsf{n}} - \frac{\partial J_\Gamma}{\partial \alpha}. \tag{4.2.19c}$$

Eqn. (4.2.19c) is used to determine β, and as at the inlet, Eqn. (4.2.11) applies.

4.2.4 Adjoint Boundary Conditions at the Overflow

At the overflow, typical primal conditions are zero pressure and zero gradient of velocity and dispersed-phase volume fraction. Therefore, we have,

$$\delta p_m = 0,\ (\mathbf{n}\cdot\nabla)\delta\mathbf{v}_m = 0 \text{ and } (\mathbf{n}\cdot\nabla)\delta\alpha = 0. \tag{4.2.20}$$

The second integral in Eqns. (4.2.6a) and (4.2.6c) therefore go to zero and, with $\delta p_m = 0$, Eqn. (4.2.6b) is identically fulfilled. The remaining terms in Eqns. (4.2.6) are the first integrals in Eqns. (4.2.6a) and (4.2.6c), which can be made to go to zero by enforcing the integrands to vanish:

$$\mathbf{n}(\mathbf{u}\cdot\rho_m\mathbf{v}_m) + \rho_m v_{m\mathsf{n}}\mathbf{u} + \mu_m(\mathbf{n}\cdot\nabla)\mathbf{u} - q\rho_m\mathbf{n} + \alpha\beta\mathbf{n} + \frac{\partial J_\Gamma}{\partial \mathbf{v}_m} = 0, \tag{4.2.21a}$$

$$\big(v_{m\mathsf{n}} - (\alpha k_1 - 1)v_{dj\mathsf{n}}\big)\beta + K(\mathbf{n}\cdot\nabla)\beta + C_1 u_{\mathsf{n}} + C_2 v_{m\mathsf{n}} + \frac{\partial J_\Gamma}{\partial \alpha} = 0. \tag{4.2.21b}$$

Decomposing Eqn. (4.2.21a) into its normal and tangential components yields:

$$\rho_m\mathbf{u}\cdot\mathbf{v}_m + \rho_m u_{\mathsf{n}} v_{m\mathsf{n}} + \mu_m(\mathbf{n}\cdot\nabla)u_{\mathsf{n}} - \rho_m q + \alpha\beta + \frac{\partial J_\Gamma}{\partial v_{m\mathsf{n}}} = 0, \tag{4.2.22a}$$

$$\rho_m v_{m\mathsf{n}}\mathbf{u}_{\mathsf{t}} + \mu_m(\mathbf{n}\cdot\nabla)\mathbf{u}_{\mathsf{t}} + \frac{\partial J_\Gamma}{\partial \mathbf{v}_{m\mathsf{t}}} = 0. \tag{4.2.22b}$$

Eqns. (4.2.21b), (4.2.22a and (4.2.22b) are used to determine β, q and $\mathbf{u}_{\mathsf{t}}$, respectively. Since u_{n} is prescribed at the inlet and underflow, the adjoint continuity equation, Eqn. (4.2.1b), is used to calculate u_{n} at the overflow as a function of u_{n} at the inlet and underflow, Φ_Γ. The boundary conditions for the adjoint variables at the overflow are summarised as:

$$\rho_m v_{m\mathsf{n}}\mathbf{u}_{\mathsf{t}} + \mu_m(\mathbf{n}\cdot\nabla)\mathbf{u}_{\mathsf{t}} = -\frac{\partial J_\Gamma}{\partial \mathbf{v}_{m\mathsf{t}}}, \tag{4.2.23a}$$

$$u_{\mathsf{n}} = \Phi_\Gamma, \tag{4.2.23b}$$

$$\big(v_{m\mathsf{n}} - (\alpha k_1 - 1)v_{dj\mathsf{n}}\big)\beta = -K(\mathbf{n}\cdot\nabla)\beta - C_1 u_{\mathsf{n}} - C_2 v_{m\mathsf{n}} - \frac{\partial J_\Gamma}{\partial \alpha}, \tag{4.2.23c}$$

$$q = \mathbf{u}\cdot\mathbf{v}_m + u_{\mathsf{n}} v_{m\mathsf{n}} + \nu_m(\mathbf{n}\cdot\nabla)u_{\mathsf{n}} + \frac{\alpha\beta}{\rho_m} + \frac{1}{\rho_m}\frac{\partial J_\Gamma}{\partial v_{m\mathsf{n}}}, \tag{4.2.23d}$$

where $\nu_m = \dfrac{\mu_m}{\rho_m}$ is the mixture kinematic viscosity. A summary of the boundary conditions for the adjoint variables is presented in Table 4.2.

Table 4.2: Adjoint boundary conditions for ducted flows.

	$\mathbf{u}_{\mathsf{t}}$	u_{n}	β	q
Inlet	zero	(4.2.10b)	zero	zero gradient
Wall	zero	(4.2.15b)	(4.2.15c)	zero gradient
Underflow	zero	(4.2.19b)	(4.2.19c)	zero gradient
Overflow	(4.2.23a)	(4.2.23b)	(4.2.23c)	(4.2.23d)

4.3 Objective Function

The objective function can be defined at the boundary and interior of the domain, Eqn. (4.1.26). However, as stated in Section 4.2, there is no contribution to the objective function from the interior of the domain, because the efficiency of an HDVS is defined in terms of the dispersed-phase mass at the boundaries of the domain (Egarr et al., 2009). In this derivation, the objective function is defined not only in terms of the dispersed-phase mass at the boundaries but also in terms of its velocity, in order to know whether it is entering, leaving or stagnating at the boundaries. Therefore, the objective function is defined as the dispersed-phase mass-flow rate at the boundaries of the domain,

$$J_\Gamma = \alpha\rho_d \mathbf{v}_d \cdot \mathbf{n}, \tag{4.3.1}$$

where $\alpha\rho_d$ is the dispersed-phase mass fraction and $\mathbf{v}_d \cdot \mathbf{n}$ is the dispersed-phase velocity normal to the boundary. For the HDVS, the boundaries of interest are the underflow and overflow so, Eqn. (4.1.26) becomes,

$$J = \int_{\Gamma_u} w_u J_u \, d\Gamma_u + \int_{\Gamma_o} w_o J_o \, d\Gamma_o, \tag{4.3.2}$$

where w_u and w_o are the weightings applied to the underflow and overflow respectively. Specifically, $w_u = 1$ and $w_o = -2$ have been used to penalise the objective function at the overflow. However, other values can be used, depending on the desired penalty for the objective function at the overflow.

Decomposing the dispersed-phase velocity into the mixture and dispersed-phase diffusion velocities, Eqn. (4.3.1) can be rewritten as,

$$\begin{aligned} J_\Gamma &= \alpha\rho_d(\mathbf{v}_m + \mathbf{v}_{dm}) \cdot \mathbf{n} \\ &= \alpha\rho_d \left(\mathbf{v}_m + \frac{\rho_c}{\rho_m}\mathbf{v}_{dj} \right) \cdot \mathbf{n} \\ &= \alpha\rho_d \left(v_\mathsf{n} + \frac{\rho_c}{\rho_m} v_{dj\mathsf{n}} \right), \end{aligned} \tag{4.3.3}$$

where $\mathbf{v}_{dj}$ is defined relative to $\mathbf{v}_{dm}$ in Eqn. (2.3.1). Substituting Eqn. (4.1.11) into Eqn. (4.3.3) and rewriting the parentheses as binomial expansions,

$$J_\Gamma = \alpha\rho_d \left(v_\mathsf{n} + (1-\alpha)^{-1} \left(1 + \frac{\alpha}{1-\alpha}\frac{\rho_d}{\rho_c} \right)^{-1} v_{dj\mathsf{n}} \right) \tag{4.3.4}$$

$$= \alpha\rho_d \left(v_\mathsf{n} + (1 + \alpha + \cdots) \left(1 - \frac{\alpha}{1-\alpha}\frac{\rho_d}{\rho_c} + \cdots \right) v_{dj\mathsf{n}} \right) \tag{4.3.5}$$

$$= \alpha\rho_d \left(v_\mathsf{n} + \left(1 + \alpha \left(1 - \frac{1}{1-\alpha}\frac{\rho_d}{\rho_c} \right) + \cdots \right) v_{dj\mathsf{n}} \right). \tag{4.3.6}$$

As $\alpha \ll 1$ and $\rho_d \approx 2\rho_c \implies \left|1 - \frac{1}{1-\alpha}\frac{\rho_d}{\rho_c}\right| < 2$ and ignoring terms containing squared and higher powers of α,

$$J_\Gamma \approx \alpha\rho_d(v_{\mathsf{n}} + v_{dj\mathsf{n}}). \tag{4.3.7}$$

In this formulation $\mathbf{v}_{dj}$ is modelled using Eqn. (2.2.6). Applying the product rule,

$$\begin{aligned}\frac{\partial J_\Gamma}{\partial \alpha} &= \rho_d\left(\alpha(-k_1 v_{dj\mathsf{n}}) + v_{\mathsf{n}} + v_{dj\mathsf{n}}\right) \\ &= \rho_d\big(v_{\mathsf{n}} - v_{dj\mathsf{n}}(\alpha k_1 - 1)\big).\end{aligned} \tag{4.3.8}$$

The derivatives of the objective function, Eqn. (4.3.1), with respect to the primal variables can now be summarised as:

$$\frac{\partial J_\Gamma}{\partial v_{m\mathsf{n}}} = \alpha\rho_d, \tag{4.3.9a}$$

$$\frac{\partial J_\Gamma}{\partial \mathbf{v}_{m\mathsf{t}}} = 0, \tag{4.3.9b}$$

$$\frac{\partial J_\Gamma}{\partial \alpha} = \rho_d\big(v_{m\mathsf{n}} - (\alpha k_1 - 1)v_{dj\mathsf{n}}\big), \tag{4.3.9c}$$

$$\frac{\partial J_\Gamma}{\partial p_m} = 0. \tag{4.3.9d}$$

Using these derivatives, the adjoint boundary conditions at the inlet reduce to:

$$\mathbf{u}_{\mathsf{t}} = 0, \tag{4.3.10a}$$

$$u_{\mathsf{n}} = 0, \tag{4.3.10b}$$

$$\beta = 0, \tag{4.3.10c}$$

$$(\mathbf{n}\cdot\nabla)q = 0. \tag{4.3.10d}$$

At the wall, there is no contribution from the objective function, so:

$$\mathbf{u}_{\mathsf{t}} = 0, \tag{4.3.11a}$$

$$u_{\mathsf{n}} = 0, \tag{4.3.11b}$$

$$-(\alpha k_1 - 1)v_{dj\mathsf{n}}\beta + K(\mathbf{n}\cdot\nabla)\beta = 0, \tag{4.3.11c}$$

$$(\mathbf{n}\cdot\nabla)q = 0. \tag{4.3.11d}$$

At the underflow:

$$\mathbf{u}_\mathsf{t} = 0, \tag{4.3.12a}$$

$$u_\mathsf{n} = 0, \tag{4.3.12b}$$

$$\big(v_{m\mathsf{n}} - (\alpha k_1 - 1)v_{dj\mathsf{n}}\big)\beta + K(\mathbf{n}\cdot\nabla)\beta = -C_2 v_{m\mathsf{n}} - w_u \frac{\partial J_\Gamma}{\partial \alpha}, \tag{4.3.12c}$$

$$(\mathbf{n}\cdot\nabla)q = 0. \tag{4.3.12d}$$

At the overflow, to satisfy the adjoint continuity equation, Eqn. (4.2.1b), $u_\mathsf{n} = 0$, so:

$$v_{m\mathsf{n}}\mathbf{u}_\mathsf{t} + \nu_m(\mathbf{n}\cdot\nabla)\mathbf{u}_\mathsf{t} = 0, \tag{4.3.13a}$$

$$u_\mathsf{n} = 0, \tag{4.3.13b}$$

$$\big(v_{m\mathsf{n}} - (\alpha k_1 - 1)v_{dj\mathsf{n}}\big)\beta + K(\mathbf{n}\cdot\nabla)\beta = -C_2 v_{m\mathsf{n}} - w_o \frac{\partial J_\Gamma}{\partial \alpha}, \tag{4.3.13c}$$

$$q = \mathbf{u}\cdot\mathbf{v}_m + \nu_m(\mathbf{n}\cdot\nabla)u_\mathsf{n} + \frac{\alpha\beta}{\rho_m} + \frac{w_o}{\rho_m}\frac{\partial J_\Gamma}{\partial v_{m\mathsf{n}}}. \tag{4.3.13d}$$

A summary of the adjoint boundary conditions, using the objective function defined in Eqn. (4.3.1), is presented in Table 4.3.

Table 4.3: Adjoint boundary conditions, using objective function Eqn. (4.3.1).

	$\mathbf{u}_\mathsf{t}$	u_n	β	q
Inlet	zero	zero	zero	zero gradient
Wall	zero	zero	(4.3.11c)	zero gradient
Underflow	zero	zero	(4.3.12c)	zero gradient
Overflow	(4.3.13a)	zero	(4.3.13c)	(4.3.13d)

4.4 Coding the Adjoint Equations

In OpenFOAM®, the first argument in the divergence operator is a flux. Hence, in the adjoint drift equation, Eqn. (4.2.1c), β is replaced using the transformation,

$$B = \frac{\rho_d}{\rho_m}\beta, \tag{4.4.1}$$

so the adjoint drift flux equations, Eqns. (4.2.1), are reformulated as:

$$-(\rho_m \mathbf{v}_m \cdot \nabla)\mathbf{u} = -\rho_m \nabla q + \nabla \mathbf{u} \cdot (\rho_m \mathbf{v}_m) + \nabla \cdot \left(2\mu_m \mathsf{D}(\mathbf{u})\right) + \alpha \nabla \beta - \aleph \rho_m \mathbf{u}, \tag{4.4.2a}$$

$$\nabla \cdot (\rho_m \mathbf{u}) = 0, \tag{4.4.2b}$$

$$-(\rho_m \mathbf{v}_m \cdot \nabla)B = -(\alpha k_1 - 1)(\rho_m \mathbf{v}_{dj} \cdot \nabla)B + \nabla \cdot \mu_m^t \nabla B + \rho_d (S_1 + S_2). \tag{4.4.2c}$$

The OpenFOAM® coding for these equations is given in Appendix B.1, and are solved using the SIMPLE algorithm (Jasak, 1996).

4.5 Coding the Adjoint Boundary Conditions

Figure 4.3 illustrates the notation at the boundary, where f represents the boundary with normal unit vector, **n**, at computational point, p. The internal cell abutting the face is denoted by c and the vector between c and p as Δ.

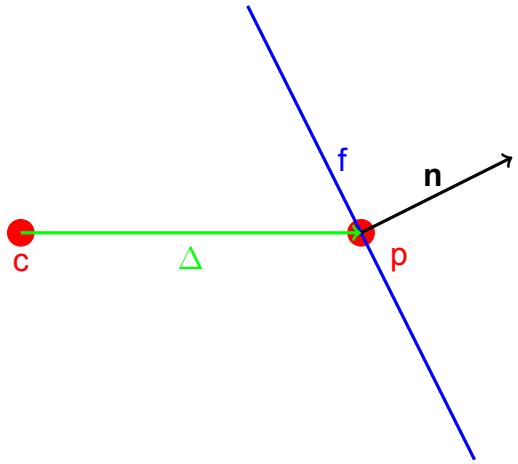


Figure 4.3: Boundary face notation.

The equations in Table 4.3 include a Neumann (fixed gradient) boundary condition of the form $(\mathbf{n} \cdot \nabla)\phi$, where ϕ represents $\mathbf{u}_t$, u_n and β. This condition is discretised using the approximation,

$$(\mathbf{n} \cdot \nabla)\phi \approx \frac{\phi_f - \phi_c}{\Delta_n}, \tag{4.5.1}$$

where ϕ_f and ϕ_c represent the values of ϕ at the boundary face and its neighbouring cell, respectively, and Δ_n is the normal distance between them. The equations for β are mixed boundary conditions, meaning they include a Dirichlet (fixed value) and Neumann boundary condition. In these cases, the Robin condition is used,

which is a blended combination of a fixed value and fixed gradient condition,

$$\phi_{\mathsf{f}} = w_V \phi_V + (1 - w_V)\phi_G \tag{4.5.2a}$$
$$= w_V \phi_V + (1 - w_V)(\phi_{\mathsf{c}} + \Delta \cdot \nabla \phi_{\mathsf{f}}) \tag{4.5.2b}$$
$$= w_V \phi_V + (1 - w_V)(\phi_{\mathsf{c}} + \Delta_{\mathsf{n}} G), \tag{4.5.2c}$$

where ϕ_V is the fixed value part, ϕ_G is the gradient-evaluated part, w_V is the weighting assigned to ϕ_V ($0 < w_V < 1$) and $G = (\mathbf{n} \cdot \nabla)\phi_{\mathsf{f}}$ is the surface-normal component of $\nabla \phi_{\mathsf{f}}$. Note that $G \approx |\nabla \phi_{\mathsf{f}}|$ because, in practice, only the surface-normal component of the gradient is prescribed.

4.5.1 Coding Adjoint Pressure at the Overflow

The adjoint pressure at the overflow is given by Eqn. (4.3.13d). Using Eqn. (4.5.1), where $u_{\mathsf{n}} = 0$ at the boundary face, Eqn. (4.3.13d) can be discretised as,

$$q = \mathbf{u} \cdot \mathbf{v}_m - \nu_m \frac{u_{\mathsf{nc}}}{\Delta_{\mathsf{n}}} + \frac{\alpha\beta}{\rho_m} + \frac{w_o}{\rho_m} \frac{\partial J_\Gamma}{\partial v_{m\mathsf{n}}}, \tag{4.5.3}$$

where u_{nc} is the magnitude of the normal component of adjoint velocity at the neighbouring cell. The OpenFOAM® coding for this equation is given in Appendix B.2.1.

4.5.2 Coding Adjoint Velocity at the Overflow

The adjoint velocity at the overflow is given by Eqns. (4.3.13a) and (4.3.13b). Using Eqn. (4.5.1), Eqn. (4.3.13a) can be rearranged as,

$$\mathbf{u}_{\mathsf{tf}} = \frac{\mathbf{u}_{\mathsf{tc}}}{1 + \Delta_{\mathsf{n}} \dfrac{v_{\mathsf{nf}}}{\nu_m}}, \tag{4.5.4}$$

where $\mathbf{u}_{\mathsf{tf}}$ and $\mathbf{u}_{\mathsf{tc}}$ are the adjoint tangential velocities at the boundary face and its neighbouring cell, respectively, and v_{nf} is the magnitude of the normal component of velocity at the boundary face. The OpenFOAM® coding for this equation is given in Appendix B.2.2.

4.5.3 Coding Beta at the Wall, Underflow and Overflow

At the wall, underflow and overflow, the boundary conditions for β are given by Eqns. (4.3.11c), (4.3.12c) and (4.3.13c), respectively. Since the adjoint drift equation is solved for B (Section 4.4), the adjoint boundary conditions should also be

solved for B, so the above equations are reformulated as:

$$-\rho_m(\alpha k_1 - 1)v_{dj\mathsf{n}}B + \mu_m^t(\mathbf{n}\cdot\nabla)B = 0, \tag{4.5.5a}$$

$$\rho_m\big(v_{m\mathsf{n}} - (\alpha k_1 - 1)v_{dj\mathsf{n}}\big)B + \mu_m^t(\mathbf{n}\cdot\nabla)B = -\rho_d\left(C_2 v_{m\mathsf{n}} + w_u\frac{\partial J_\Gamma}{\partial\alpha}\right), \tag{4.5.5b}$$

$$\rho_m\big(v_{m\mathsf{n}} - (\alpha k_1 - 1)v_{dj\mathsf{n}}\big)B + \mu_m^t(\mathbf{n}\cdot\nabla)B = -\rho_d\left(C_2 v_{m\mathsf{n}} + w_o\frac{\partial J_\Gamma}{\partial\alpha}\right). \tag{4.5.5c}$$

These are formalised,

$$E_1\phi + \mu_m^t(\mathbf{n}\cdot\nabla)\phi = E_2, \tag{4.5.6}$$

discretised, using Eqn. (4.5.1),

$$E_1\phi_{\mathsf{f}} + \mu_m^t\frac{\phi_{\mathsf{f}} - \phi_{\mathsf{c}}}{\Delta_{\mathsf{n}}} = E_2, \tag{4.5.7}$$

and rearranged as,

$$\phi_{\mathsf{f}} = \left(1 - \frac{1}{1 + \dfrac{\mu_m^t}{E_1\Delta_{\mathsf{n}}}}\right)\left(\phi_{\mathsf{c}} + \Delta_{\mathsf{n}}\frac{E_2}{\mu_m^t}\right). \tag{4.5.8}$$

Comparing Eqns. (4.5.2c) and (4.5.8), we see that:

$$\phi_V = 0, \tag{4.5.9a}$$

$$G = \frac{E_2}{\mu_m^t}, \tag{4.5.9b}$$

$$w_V = \frac{1}{1 + \dfrac{\mu_m^t}{E_1\Delta_{\mathsf{n}}}}. \tag{4.5.9c}$$

Hence, the boundary conditions for B deduced from Eqn. (4.5.8) are listed below. At the wall:

$$E_1 = -\rho_m(\alpha k_1 - 1)v_{dj\mathsf{n}}, \tag{4.5.10a}$$

$$E_2 = 0. \tag{4.5.10b}$$

At the underflow:

$$E_1 = \rho_m\big(v_{m\mathsf{n}} - (\alpha k_1 - 1)v_{dj\mathsf{n}}\big), \tag{4.5.11a}$$

$$E_2 = -\rho_d\left(C_2 v_{m\mathsf{n}} + w_u\frac{\partial J_\Gamma}{\partial\alpha}\right). \tag{4.5.11b}$$

At the overflow:

$$E_1 = \rho_m \big(v_{m\mathsf{n}} - (\alpha k_1 - 1) v_{dj\mathsf{n}} \big), \tag{4.5.12a}$$

$$E_2 = -\rho_d \left(C_2 v_{m\mathsf{n}} + w_o \frac{\partial J_\Gamma}{\partial \alpha} \right). \tag{4.5.12b}$$

The OpenFOAM® coding for these boundary conditions is given in Appendices B.2.3, B.2.4 and B.2.5, respectively.

Chapter 5

Optimisation Study

5.1 Objective

The methodology of Chapter 4 is applied to a simplified HDVS, with the objective of finding where to modify its internal shape for enhanced performance. The methodology is applied using typical operating conditions to find an optimal design, and the performance of the initial and optimal designs are compared using three different operating conditions. Only one optimal design is attempted, because the focus is on demonstrating a proof of concept rather than performing a complete study. CFD simulations are notoriously time consuming to run and, in order to facilitate the process, some modifications have been made to the initial HDVS geometry and to the simulation parameters, which are explained in Section 5.2.

5.2 Method

The flow of wastewater is modelled using the drift flux equations, which are solved on the initial design to a quasi-steady state. The adjoint drift flux equations are then solved on the initial design, using the quasi-steady state solution of the drift flux equations, in order to calculate the sensitivity of the objective function with respect to the design parameter. The objective function is defined as the dispersed-phase mass-flow rate at the underflow minus twice the dispersed-phase mass-flow rate at the overflow, and the design parameter is defined as the cell porosity, introduced into the momentum equation as a sink term. The shape of the HDVS is modified using the porosity, by converting cells whose porosity exceeds a critical value into walls.

5.2.1 Geometry and Mesh

As mentioned in Section 3.4 and discussed in a personal communication with Tukovic (2015), a structured, hexahedral mesh is more likely to reduce simulation run times, while producing accurate results, than an unstructured tetrahedral mesh, because the solution of the flow equations is improved when the non-orthogonality of neighbouring cells, Figure 3.8, is reduced. Therefore, a hexahedral mesh is produced and, as it is more complex to generate than a tetrahedral mesh, a simplified HDVS geometry is used, containing none of the internal components described in Chapter 3.

In order to create a structured mesh, the number of nodes across a plane must be equal. An O-H topology was generated in the circular inlet pipe, using Pointwise®, to create a structured mesh, however where the inlet pipe met the HDVS cylinder, Figure 5.1b, there were 11 nodes across the diameter of the pipe but only 5 nodes across the equivalent topology of the cylinder. The process of reducing the number of nodes across the pipe and, therefore, creating a structured mesh at the intersection of the inlet pipe and cylinder is demonstrated in Figure 5.1.

The completed mesh was validated using OpenFOAM®'s utility for checking mesh validity. The results in Appendix C show that, though $\theta > 70°$ at 89 faces, the mesh was satisfactory and the key statistics are summarised in Table 5.1. A visual inspection of the mesh, using maximum included angle[1], revealed that the majority of these cells were located at the intersection of the inlet pipe and HDVS cylinder. This was unavoidable because, where the inlet pipe intersected the side of the cylinder, the angle between the inlet pipe and cylinder wall was very small. Nevertheless, the mesh was approved because the equations of flow could be solved in OpenFOAM®.

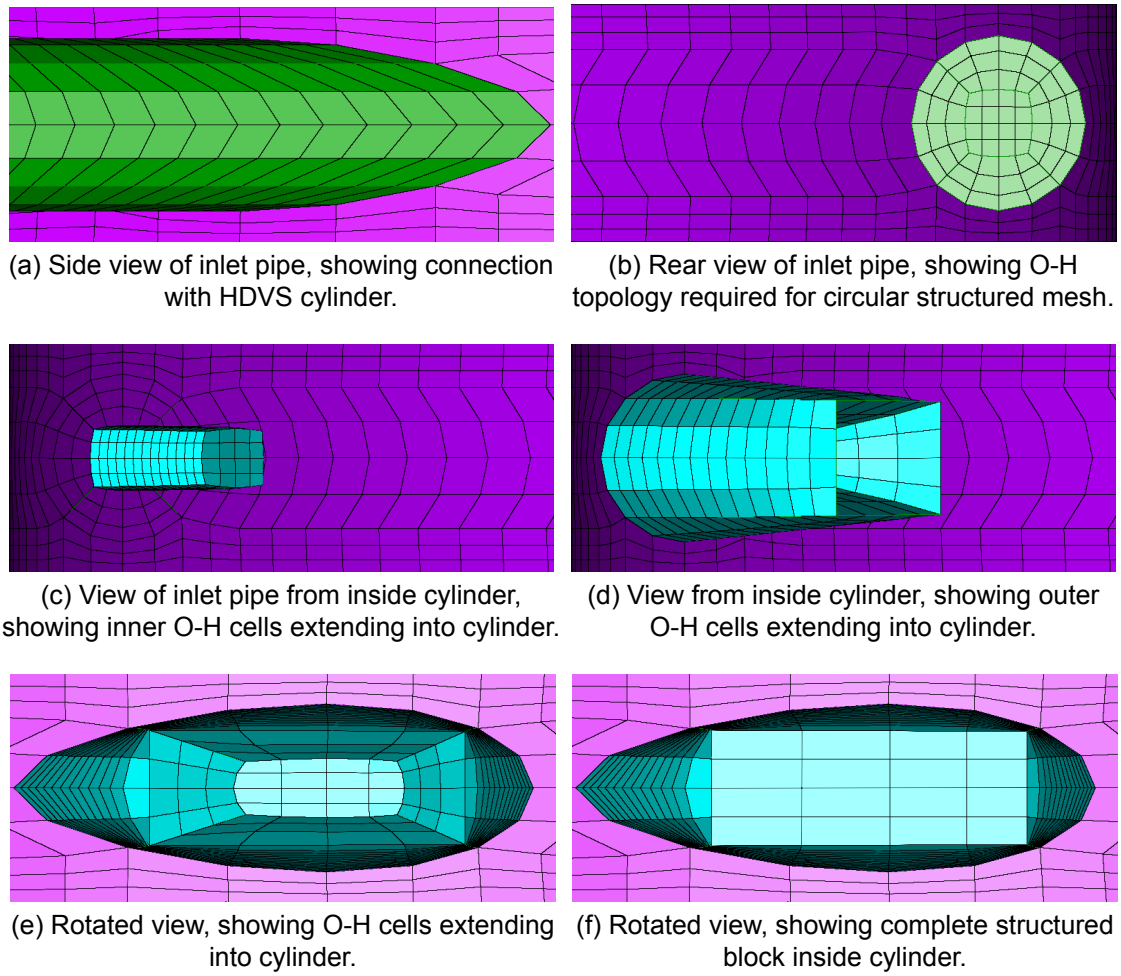
Table 5.1: Summary statistics for structured mesh.

	Structured mesh
Number of cells	1.8 million
Max non-orthogonality	76°

5.2.2 Boundary and Initial Conditions

Three test cases were set-up, each using a different combination of values for hydraulic loading rate, $\mathcal{L}$, and dispersed-phase volume fraction at the inlet, α_{in}, Table 5.2, where hydraulic loading rate is defined in Section 3.1.2 as the flow rate (l/s) per settling area (m^2) of the device.

[1] Maximum included angle in Pointwise® $\approx \theta + 90°$



(a) Side view of inlet pipe, showing connection with HDVS cylinder.

(b) Rear view of inlet pipe, showing O-H topology required for circular structured mesh.

(c) View of inlet pipe from inside cylinder, showing inner O-H cells extending into cylinder.

(d) View from inside cylinder, showing outer O-H cells extending into cylinder.

(e) Rotated view, showing O-H cells extending into cylinder.

(f) Rotated view, showing complete structured block inside cylinder.

Figure 5.1: Process for creating structured mesh at intersection of circular inlet pipe and HDVS cylinder.

Table 5.2: Differentiation of cases 1, 2 and 3 according to $\mathcal{L}$ and α_{in}.

	$\mathcal{L}$ (l/s/m^2)	α_{in} (%)
Case 1	3	0.2
Case 2	3	0.1
Case 3	2	0.2

The inlet velocity was calculated from the hydraulic loading rate and the underflow velocity was set to simulate a pump extracting sludge there at one tenth the inflow rate. Therefore in these cases, $\mathcal{L} = 3$ l/s/m^2 corresponds to an inlet velocity of 1.38 m/s and underflow velocity of 0.05 m/s, and $\mathcal{L} = 2$ l/s/m^2 corresponds to an inlet velocity of 0.92 m/s and underflow velocity of 0.04 m/s. The other primal boundary conditions are given in Table 4.1. In all three cases, the initial conditions over the interior of the domain were zero velocity, zero p_{rgh}[2] and $\alpha = \alpha_{in}$.

The adjoint boundary conditions are given in Table 4.3, with the initial adjoint conditions over the interior of the domain being $(u_x, u_y, u_z, q, \beta) = (1, 1, 1, 0, 0)$.

[2] p_{rgh} is total pressure minus hydrostatic pressure

5.2.3 Settling Velocity and Viscosity

Settling velocity can be modelled using three equations: Eqns. (2.2.5) and (2.2.11) in hindered settling conditions, and Eqn. (2.2.7) in discrete particle, flocculent and hindered settling conditions. Eqn. (2.2.11) has been shown to yield good predictions of the velocity and α distributions, with the coefficient and exponent values taken from Dahl (1993). However, in order to accelerate the process of testing the methodology, the coefficient was multiplied by 100, creating a settling velocity of approximately 6 cm/s, which is within the range of values for grit recommended by Herrick et al. (2015), approximately 1 cm/s, and Arceivala and Asolekar (2008), approximately 10 cm/s. Therefore, in order to accelerate the process, the settling velocity of grit was modelled, using Eqn. (2.2.11), with the coefficient and exponent values taken from Dahl (1993) and the coefficient value multiplied by 100.

Eqn. (2.2.11) was also used to model viscosity, with the coefficient and exponent values taken from Dahl (1993). The coefficients, a, and exponents, b, used in this study are presented in Table 5.3.

Table 5.3: Values of a and b used in Eqn. (2.2.11) to model settling velocity and viscosity in this project.

	a	$b\,(\alpha_{in} = 0.1\%)$	$b\,(\alpha_{in} = 0.2\%)$
Yield stress (kg/m/s^2)	5.5469×10^{-5}	1050.80	951.25
Plastic viscosity (kg/m/s)	2.3143×10^{-4}	179.26	179.26
Settling velocity (m/s)	-2.198×10^{-1}	-285.84	-285.84

5.2.4 Laminar Flow

As shown in Section 3.3.3, the flow is turbulent. However, the author was unable to run `driftFluxFoam`[3] with a turbulence model due to convergence issues. Nevertheless, Figure 3.12 demonstrates that, where no turbulence model was used, the velocity profile consistently overpredicted the experimental values and, as only the velocity profile is required to calculate the relative topological sensitivities, laminar flow could be used. The added advantage of not using a turbulence model is that the simulation run times are significantly reduced.

[3] OpenFOAM®'s application for solving the drift flux equations

5.2.5 Convergence

Though the drift flux equations include a temporal term, the objective was to see if the solution converged to a quasi-steady state, by inspecting whether the α and velocity fields and the dispersed-phase mass-flow rates at the outlets changed in time. Two techniques were used to determine whether this objective was achieved: the first an animation of the simulation, focusing on the α and velocity fields, and the second a time series plot of the dispersed-phase mass-flow rates at the outlets. These techniques were used in conjunction; the animation was used to provide an indication of whether the simulation had reached a quasi-steady state and the plot was used for verification.

As the adjoint drift flux equations do not include a temporal term, convergence was determined by inspecting the solution fields of the adjoint variables and a plot of their residuals.

5.2.6 Performance

Egarr et al. (2009) define the performance of an HDVS in terms of its solids' removal efficiency,

$$100\frac{V_E}{V_T}, \tag{5.2.1}$$

where V_E is the volume of particles remaining in the HDVS at the end of a test and V_T is the volume of particles released into the system. A possible bias in this definition is that, while the HDVS is being filled with particles, the device is 100% efficient and, therefore, the performance measure is distorted by the time taken for the particles released into the system to reach the overflow. To counteract this bias, the performance, $\mathcal{P}$, of an HDVS in this study is defined in terms of its rate of solids' removal efficiency,

$$\mathcal{P} = 100\left(1 - \frac{\mathcal{D}_o}{\mathcal{D}_{in}}\right), \tag{5.2.2}$$

where $\mathcal{D}_o$ is the dispersed-phase mass-flow rate at the overflow and $\mathcal{D}_{in}$ is the dispersed-phase mass-flow rate at the inlet, recorded once the system has reached a quasi-steady state. The dispersed-phase mass-flow rate at a boundary is defined as,

$$\mathcal{D} = \int_\Gamma \alpha\rho_d \mathbf{v}_d \cdot \mathbf{n}\, d\Gamma, \tag{5.2.3}$$

where $\alpha\rho_d$ is the dispersed-phase mass fraction and $\mathbf{v}_d \cdot \mathbf{n}$ is the dispersed-phase velocity normal to the boundary, Γ.

5.2.7 Scalability Study

A scalability study was performed to determine the optimum number of cores (processors) to use per simulation. The computer nodes used for running simulations had the following specifications:

Table 5.4: Computer node specifications used for simulations.

	Node 1	Node 2
CPUs	4x Intel Xeon E5-4620	2x Intel Xeon E5-2698 v3
Cores	8 @ 2.20 GHz	16 @ 2.30 GHz
RAM	132 GB	132 GB

Communication between the nodes was not good enough to use more than one node per simulation, so the study was restricted to one node, by solving the drift flux equations for a simulation run of 10 seconds, using 1, 2, 4, 8, 16 and 32 processors. Figure 5.2 shows the run time (hours) for each simulation, which as expected decreased when more processors were used. Figure 5.3 shows the ratio of simulation time to run time (S:R ratio) for each simulation, and an extrapolated straight line from the origin through the S:R ratio for one processor. It indicates an approximately linear relationship for the simulations using 1, 2, 4 and 32 processors, and a less than linear relationship for the simulations using 8 and 16 processors. The latter is probably due to the processors that were selected, for example, the communication between 8 or 16 processors over 4 CPUs would be slower than on 1 CPU. Nevertheless, the approximately linear scale up to 32 processors demonstrated that 32 processors should be used, with approximately 56,000 cells per processor.

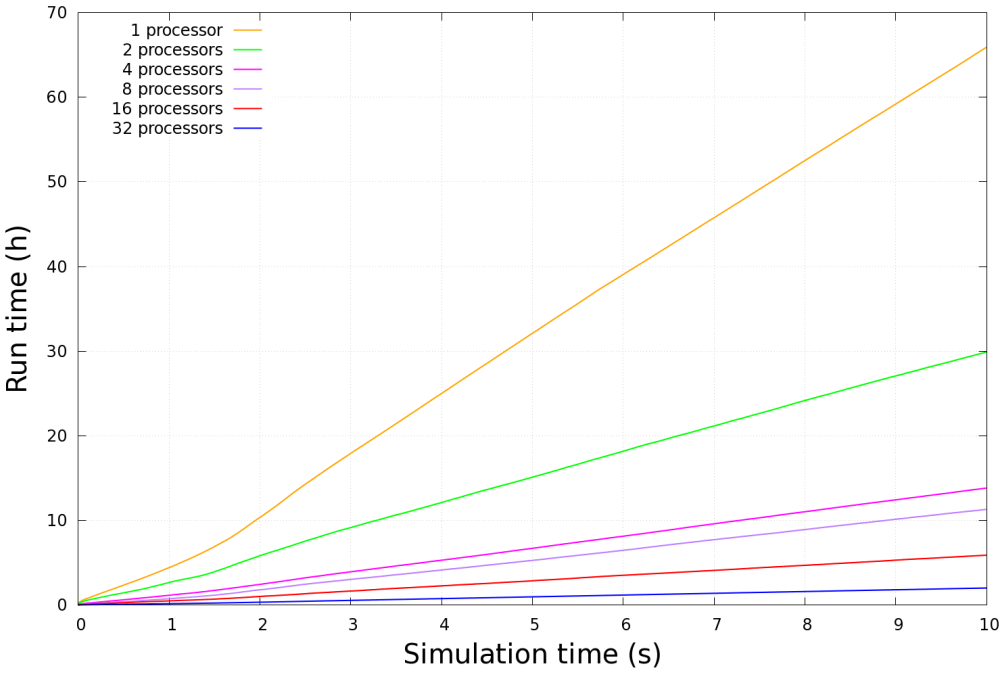


Figure 5.2: Time series plot of run time for each simulation.

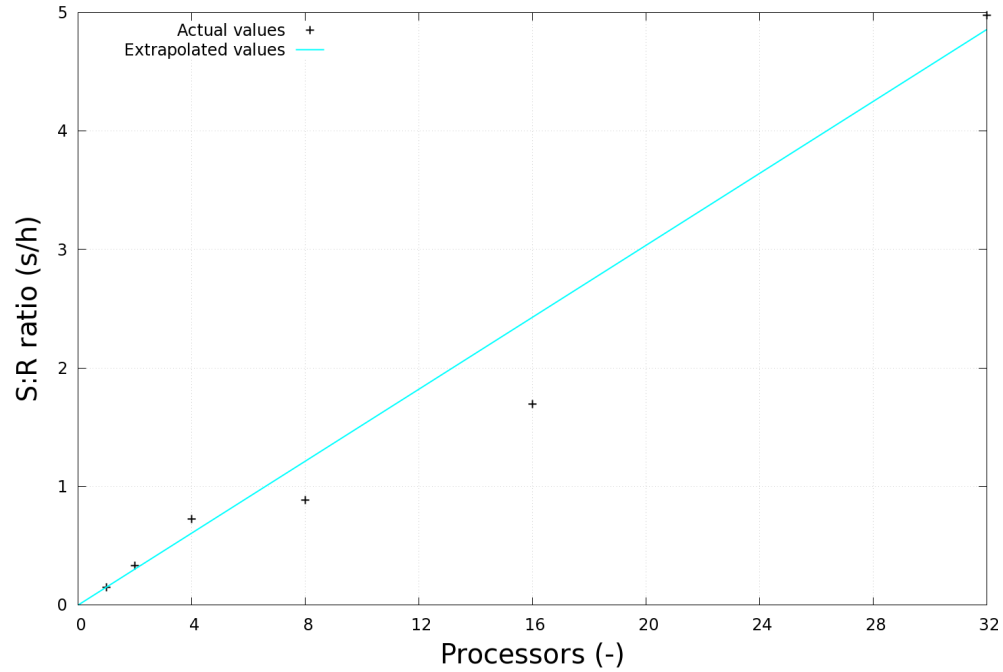


Figure 5.3: Plot of S:R ratio for each simulation, and extrapolated straight line from origin through S:R ratio for one processor.

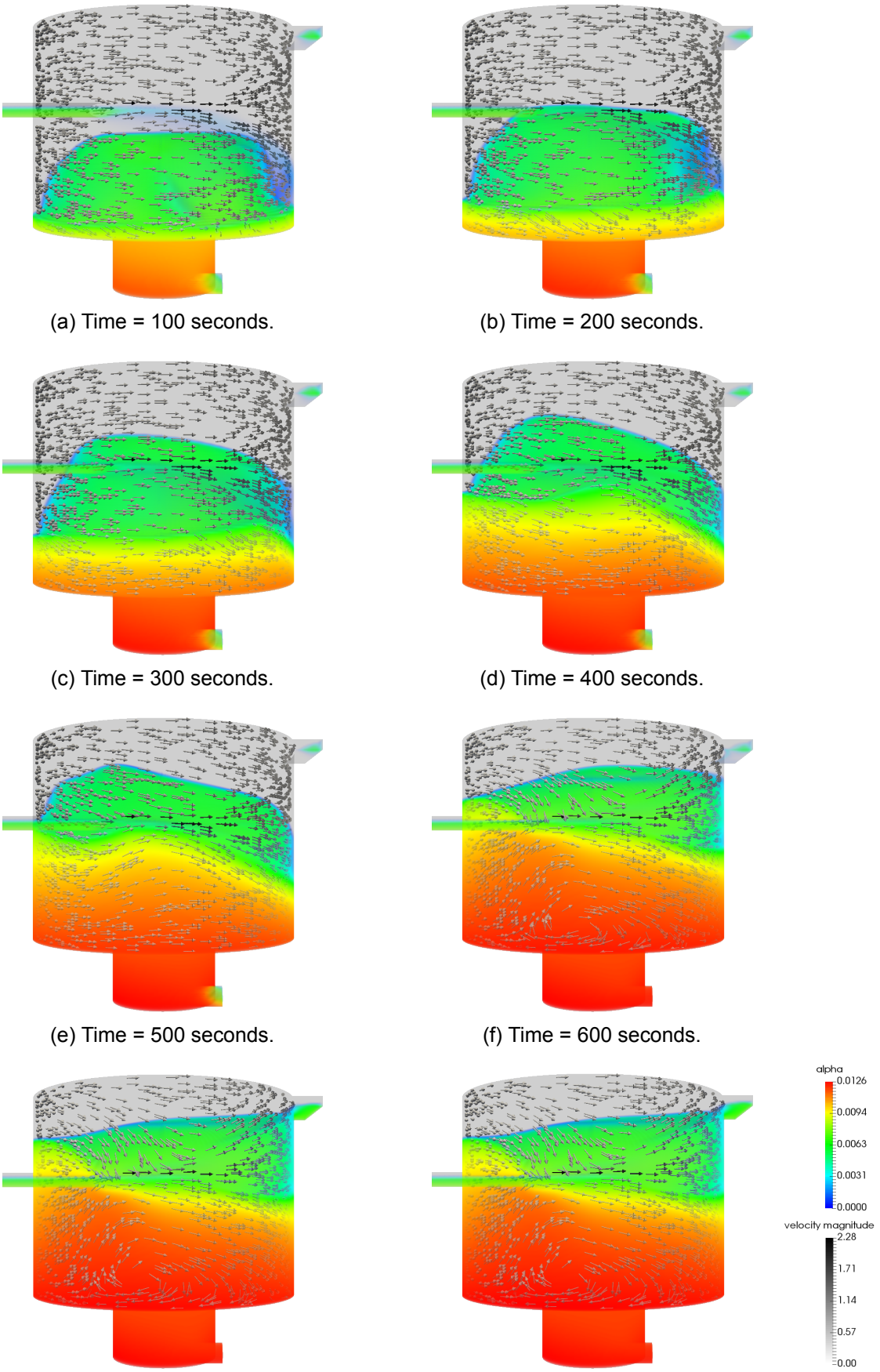
5.3 Results

5.3.1 Finding an Optimal Design: Case 1

Initial Design

The drift flux equations were solved on the initial design for a simulation run of 1,000 seconds, using the boundary and initial conditions for Case 1, Section 5.2.2. The images in Figure 5.4 are frames from an animation of the simulation, taken every 100 seconds up to 800 seconds, when the animation ceased to change. They show that, as the initial conditions are only estimates, the initial solutions are transient, while the velocity and pressure fields fluctuate and the dispersed-phase settles and then fills the HDVS. For example, Figure 5.4a shows the simulation after 100 seconds, with the dispersed-phase having occupied only the lower portion of the HDVS, where the visibility of the grey HDVS walls indicates an absence of the dispersed-phase. As the simulation progressed, the dispersed-phase gradually filled the HDVS, Figures 5.4b to 5.4f, until it reached the overflow in Figure 5.4g. Once at the overflow, the fluctuations in the α and velocity fields gradually diminished (minimal difference between 700 and 800 seconds, Figures 5.4g and 5.4h, respectively) and, then, ceased from approximately 800 seconds onwards. In all figures, it is worth noting that there is very little movement where the sludge has settled, indicated by the lack of velocity vectors there, and that there is a bifurcation of flow, adjacent to the wall opposite the end of the inlet pipe, where some of the larger velocity vectors point towards the overflow and, hence, transport material there.

Figure 5.5 is a time series plot of the dispersed-phase mass-flow rates at the outlets. It shows that, initially, the dispersed-phase mass-flow rate increased steadily at the underflow, while the dispersed-phase settled, and then increased sharply between 500 and 600 seconds. This is confirmed in Figures 5.4e and 5.4f, which show that α increased significantly at the underflow between 500 and 600 seconds. At the overflow, the dispersed-phase mass-flow rate was almost zero in Figure 5.5 and then increased, while the dispersed-phase filled the HDVS. After approximately 800 seconds, both dispersed-phase mass-flow rates remained almost constant, confirming that the simulation had reached a quasi-steady state. Note that though they were approximately equal, Figure 5.4h shows that their volume fractions were considerably different; this is due to the different flow rates at the outlets.



(a) Time = 100 seconds.

(b) Time = 200 seconds.

(c) Time = 300 seconds.

(d) Time = 400 seconds.

(e) Time = 500 seconds.

(f) Time = 600 seconds.

(g) Time = 700 seconds.

(h) Time = 800 seconds.

Figure 5.4: Depiction of α and velocity fields on initial design for case 1.

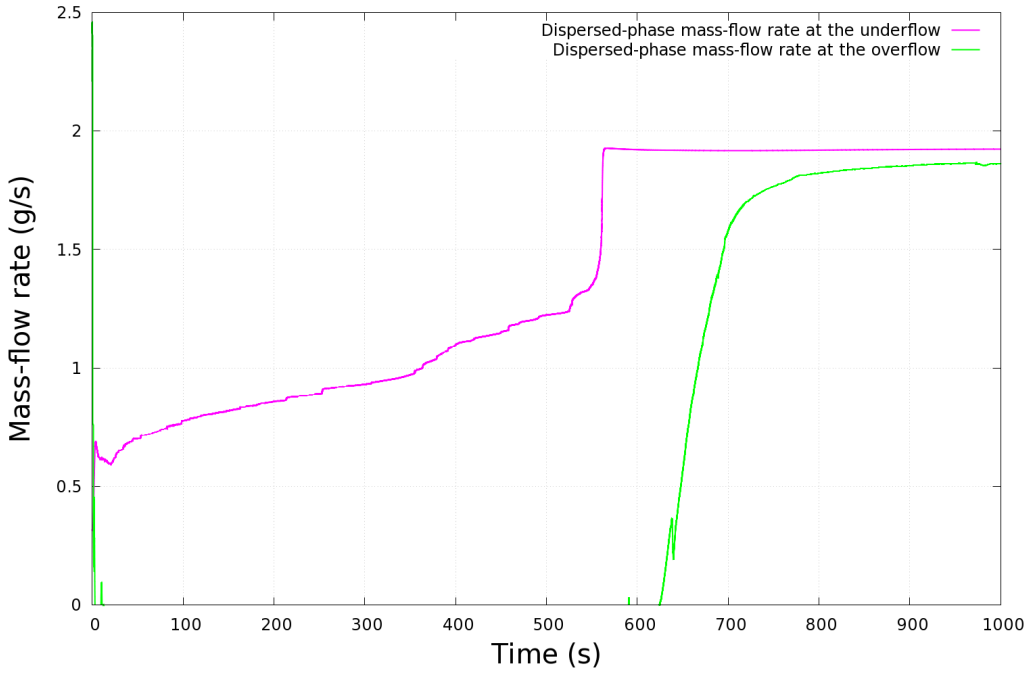


Figure 5.5: Time series plot of dispersed-phase mass-flow rates at outlets on initial design for case 1.

The adjoint drift flux equations were then solved for 50,000 iterations, using the quasi-steady state solution of the drift flux equations and the adjoint boundary and initial conditions, Section 5.2.2. Figures 5.6, 5.7 and 5.8 show the final solutions of adjoint velocity, adjoint pressure and β, respectively, and Figure 5.9 is a plot of their residuals versus iteration number. As the adjoint variables are Lagrange multipliers, not physical quantities, there is no *a priori* knowledge of what would constitute realistic values. However, as these values in the final solution are all reasonably small and their final residuals are all less than 0.001, it is concluded that the simulation reached a converged solution. It is notable that the values of the adjoint velocity are very small, implying a stagnation of the adjoint flow, which is not surprising, given the normal component of the adjoint velocity at the inlet and outlets being zero.

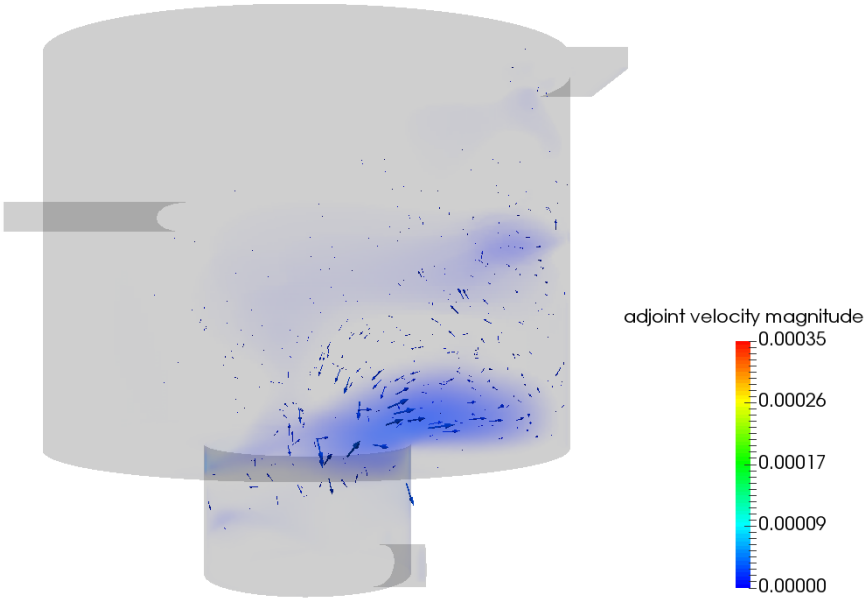


Figure 5.6: Depiction of adjoint velocity field after 50,000 iterations on initial design for case 1.

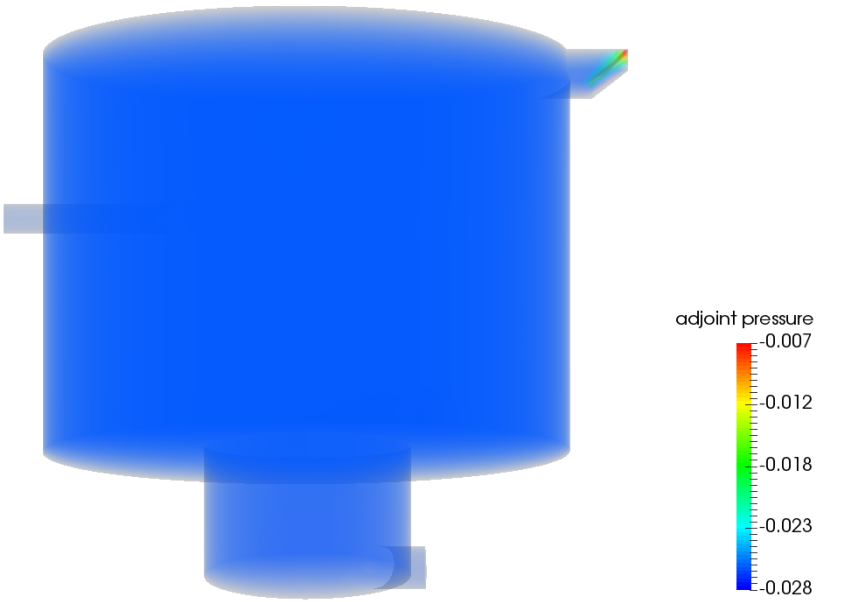


Figure 5.7: Depiction of adjoint pressure field after 50,000 iterations on initial design for case 1.

The porosity was calculated, by applying the solution of the primal and adjoint drift flux equations to Eqn. (4.1.51), and the results are presented in Figure 5.10. The highest values are adjacent to the wall opposite the end of the inlet pipe. There are also striations in the direction of the velocity vectors, Figure 5.4, which are due to the dot product of the primal and adjoint velocities in the porosity equation. For a range of critical porosities, $\aleph_{crit}$, cells whose porosity exceeded the

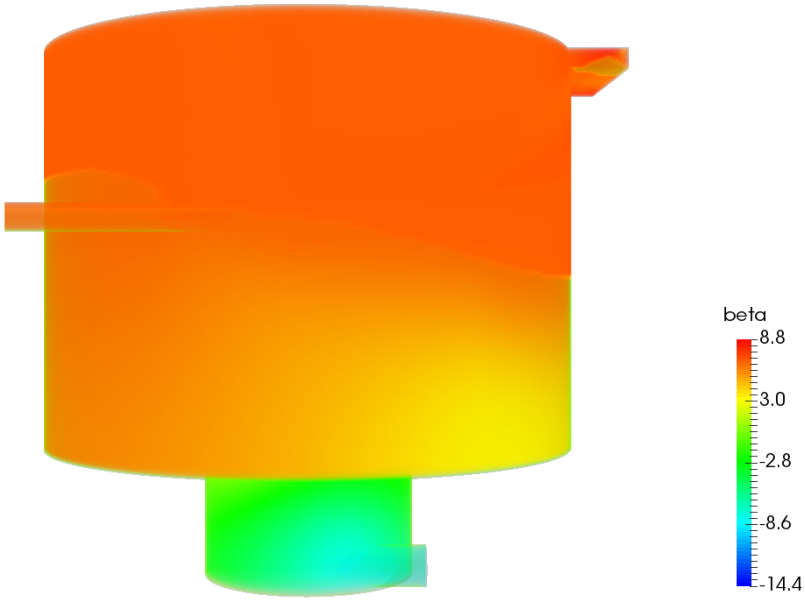


Figure 5.8: Depiction of β field after 50,000 iterations on initial design for case 1.

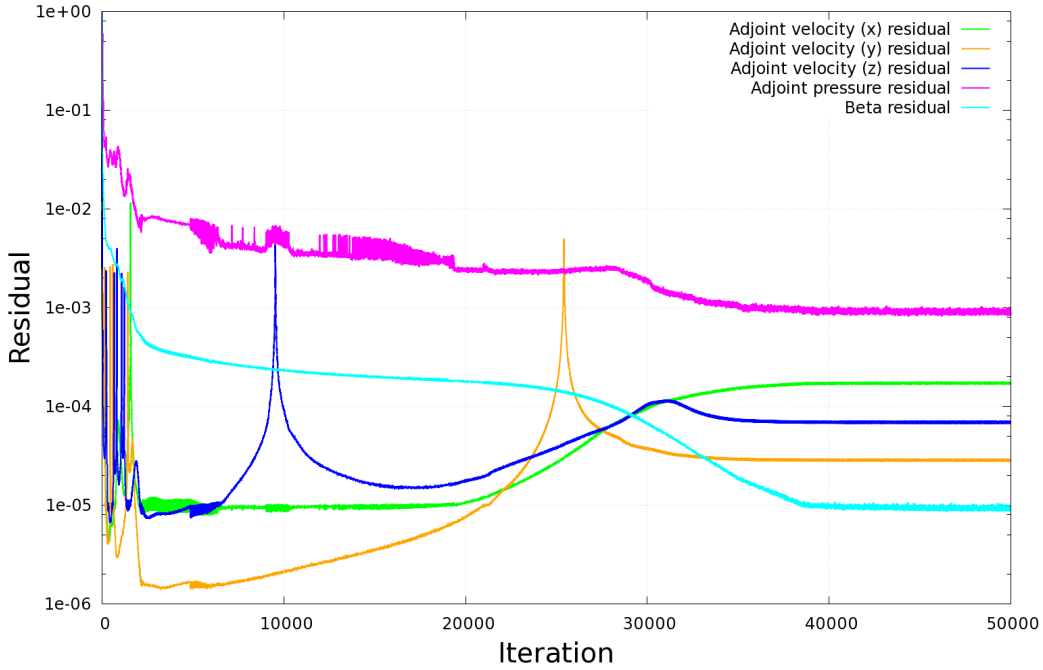


Figure 5.9: Plot of adjoint variable residuals versus iteration steps on initial design for case 1.

critical porosity were converted into walls, to create a series of modified designs. On each of these designs, the drift flux equations were solved, for a simulation run of 100 seconds, using the quasi-steady state solution on the initial design as the initial conditions. The objective was to see if the simulations reached a quasi-steady state and, if so, to calculate the associated objective function. Figure 5.11 is a plot of the objective function versus critical porosity, which shows the objective function was maximised on the design, using $\aleph_{crit} = 5$. For the remainder of

the chapter, this design is referred to as the optimal design.

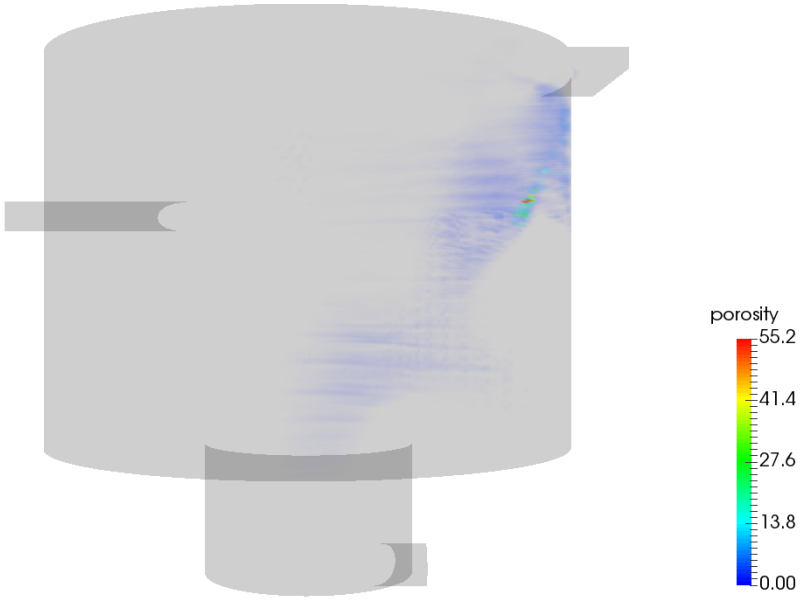


Figure 5.10: Depiction of porosity field on initial design for case 1.

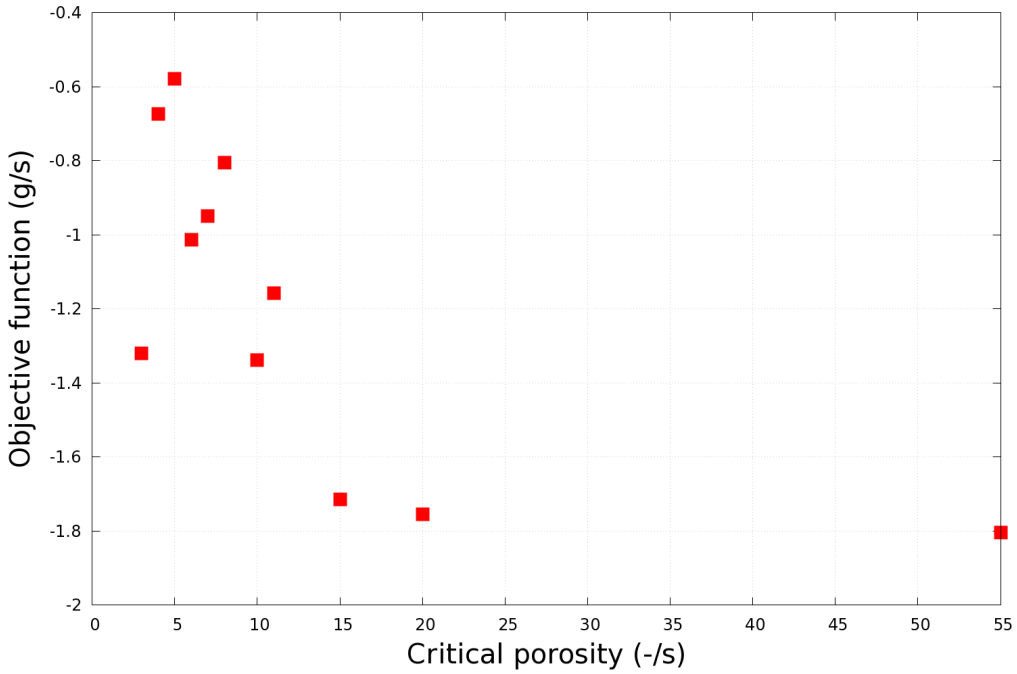


Figure 5.11: Plot of objective function versus critical porosity for case 1.

Optimal Design

A time series plot of the dispersed-phase mass-flow rates at the outlets was used to verify that the simulation had reached a quasi-steady state on this optimal design, Figure 5.12. Initially, the dispersed-phase mass-flow rate fluctuated through the overflow, while the solutions to the drift flux equations adjusted to the new design. After approximately 80 seconds, both dispersed-phase mass-flow rates remained almost constant, verifying that the simulation had reached a quasi-steady state. Figure 5.13 shows the quasi-steady state solution of the α and velocity fields, and though it looks similar to the quasi-steady state solution on the initial design, the dispersed-phase mass-flow rate at the overflow is significantly reduced: 1.25 g/s for the optimal design compared with 1.87 g/s for the initial design. Using Eqn. (5.2.2), the performance of the initial and optimal designs are 45% and 63%, respectively, Table 5.5, which represents a 40% improvement of the optimal design compared with the initial design.

Table 5.5: Performance and dispersed-phase mass-flow rates at inlet and outlets of initial and optimal designs for case 1.

	Initial	Optimal
Dispersed-phase mass-flow rate at inlet (g/s)	3.40	3.40
Dispersed-phase mass-flow rate at underflow (g/s)	1.92	1.93
Dispersed-phase mass-flow rate at overflow (g/s)	1.87	1.25
Performance	45%	63%

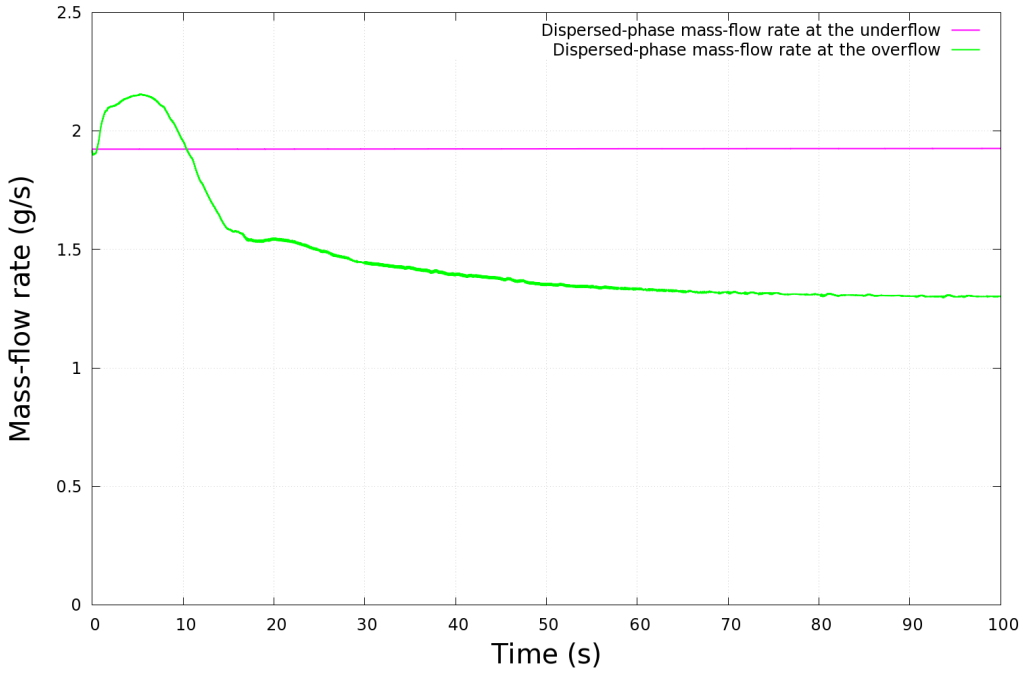


Figure 5.12: Time series plot of dispersed-phase mass-flow rates at outlets on optimal design for case 1.

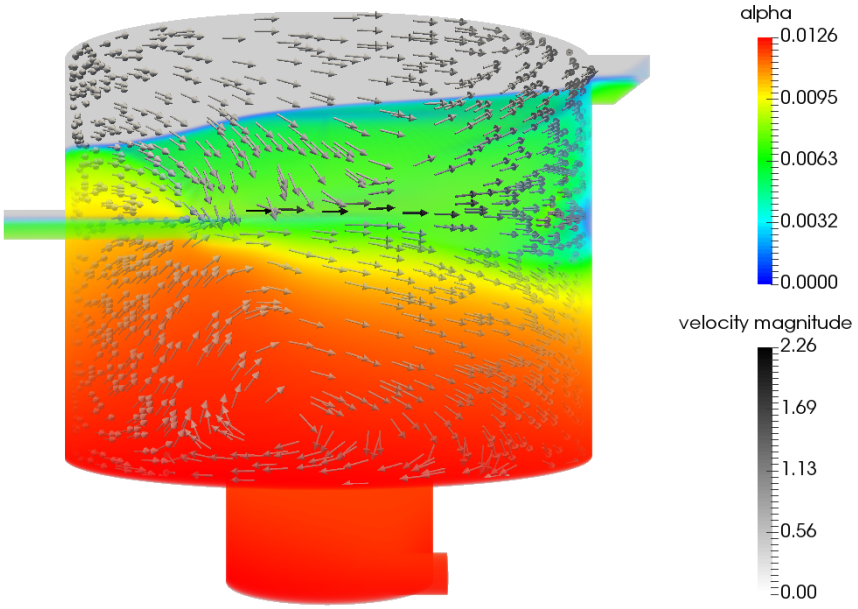


Figure 5.13: Quasi-steady state solution of α and velocity fields on optimal design for case 1.

The plan and elevation views of the optimal design are presented in Figures 5.14 and 5.15, respectively. Two things are noteworthy about the difference between the initial and optimal designs: the first is that the HDVS capacity has been reduced by less than 0.03% (132.02 litres for the initial design compared with 131.99 litres for the optimal design), and the second is that all of the converted

wall cells are adjacent to an existing wall, opposite the end of the inlet pipe and extended towards the overflow. The significance of these changes is that the capacity of the HDVS is almost unaffected and that manufacture of the optimal design would be relatively straight-forward, because none of the components are suspended inside the domain.

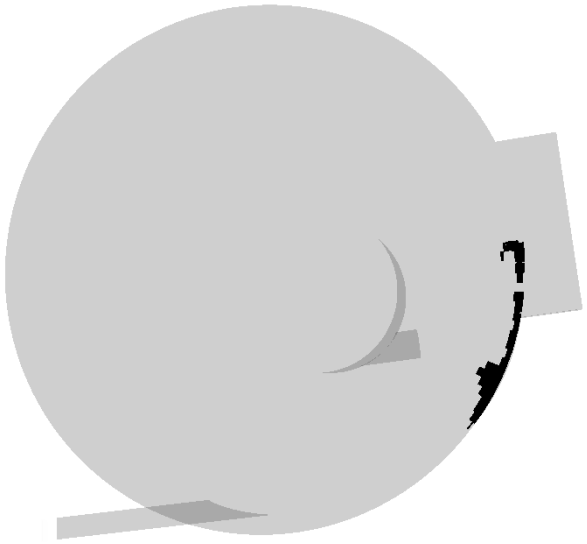


Figure 5.14: Plan view of optimal design for case 1, indicating converted wall cells adjacent to an existing wall.

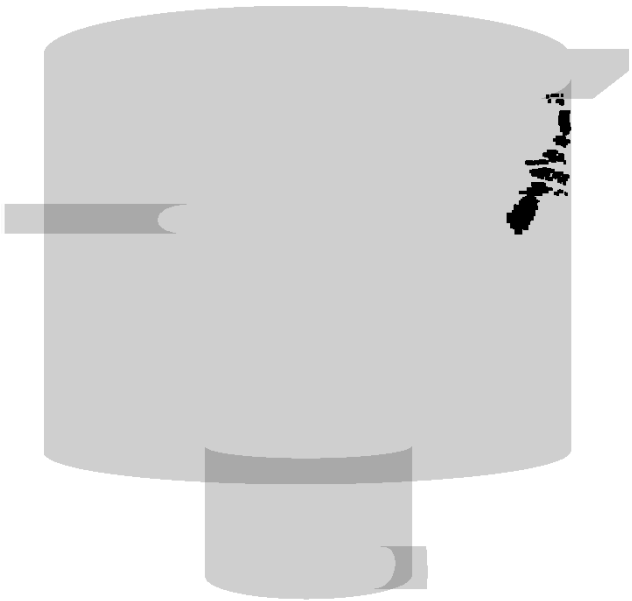


Figure 5.15: Elevation view of optimal design for case 1, indicating converted wall cells opposite the end of inlet pipe and extended towards the overflow.

5.3.2 Sensitivity Study

Case 2

The drift flux equations were solved on the initial design for a simulation run of 3,000 seconds, using the same initial and boundary conditions as for case 1, with the exception of the dispersed-phase volume fraction at the inlet, which was reduced from 0.2% to 0.1%. Figure 5.16 shows the dispersed-phase mass-flow rate at the outlets; it is noticeable that the dispersed-phase mass-flow rate at the overflow is significantly less than for case 1, due to the reduced dispersed-phase volume fraction at the inlet. Figure 5.17 shows the α and velocity fields after 3,000 seconds, confirming a smaller amount of dispersed-phase at the overflow compared with case 1.

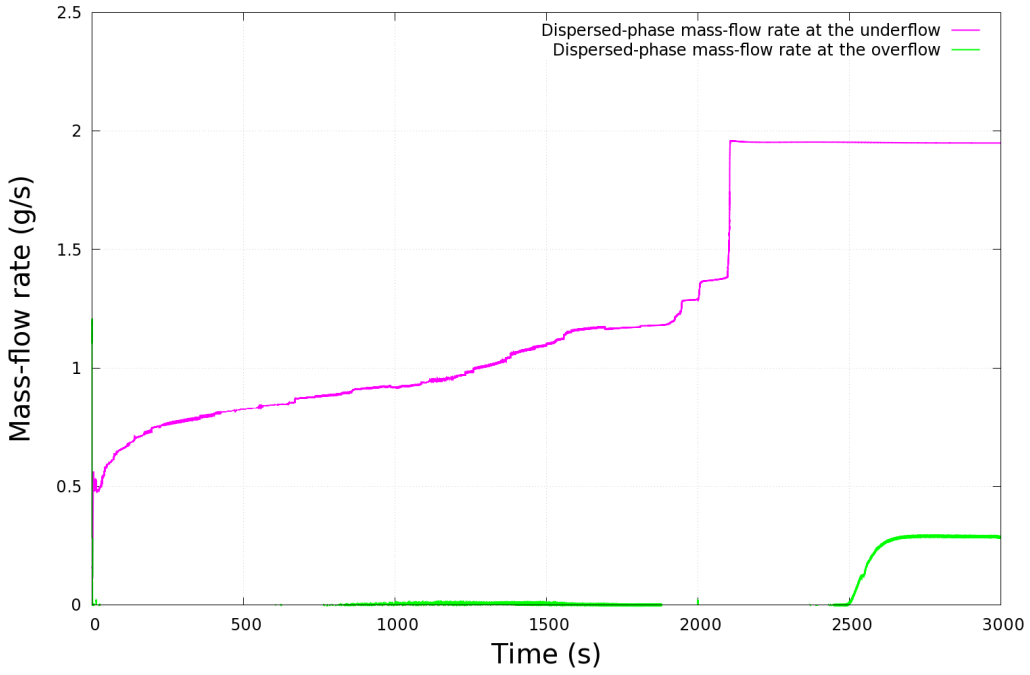


Figure 5.16: Time series plot of dispersed-phase mass-flow rates at outlets on initial design for case 2.

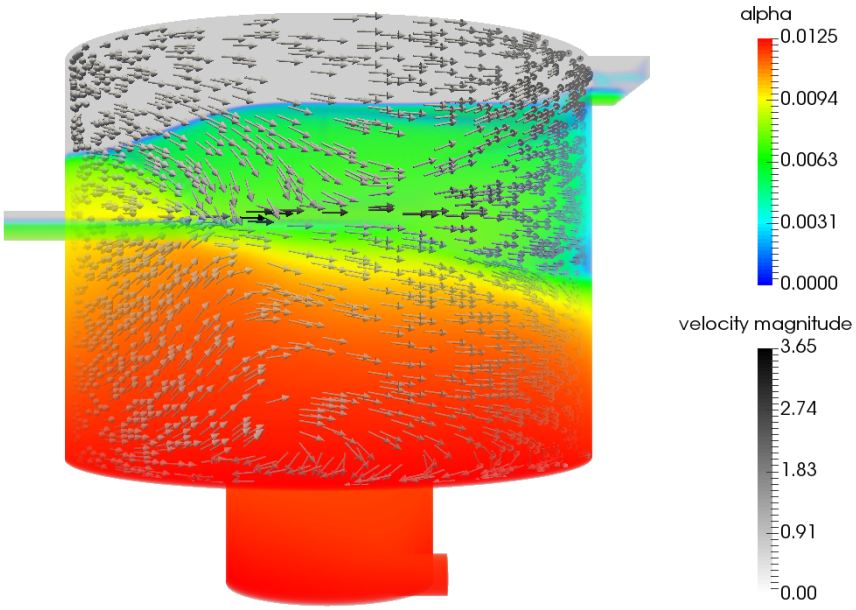


Figure 5.17: Depiction of α and velocity fields after 3,000 seconds on initial design for case 2.

The drift flux equations were then solved on the optimal design for a simulation run of 200 seconds, using the quasi-steady state solution on the initial design as the initial conditions. Figure 5.18 shows the dispersed-phase mass-flow rate at the outlets and it is noticeable that at the overflow the mass-flow rate is zero after 200 seconds. Figure 5.19 shows the α and velocity fields after 200 seconds and it is evident that the dispersed-phase does not reach the overflow. Using Eqn. (5.2.2), the performance of the initial and optimal designs are 86% and 100%, respectively, Table 5.6, which represents a 16% improvement of the optimal design compared with the initial design.

Table 5.6: Performance and dispersed-phase mass-flow rates at inlet and outlets of initial and optimal designs for case 2.

	Initial	Optimal
Dispersed-phase mass-flow rate at inlet (g/s)	1.70	1.70
Dispersed-phase mass-flow rate at underflow (g/s)	1.91	1.92
Dispersed-phase mass-flow rate at overflow (g/s)	0.24	0.00
Performance	86%	100%

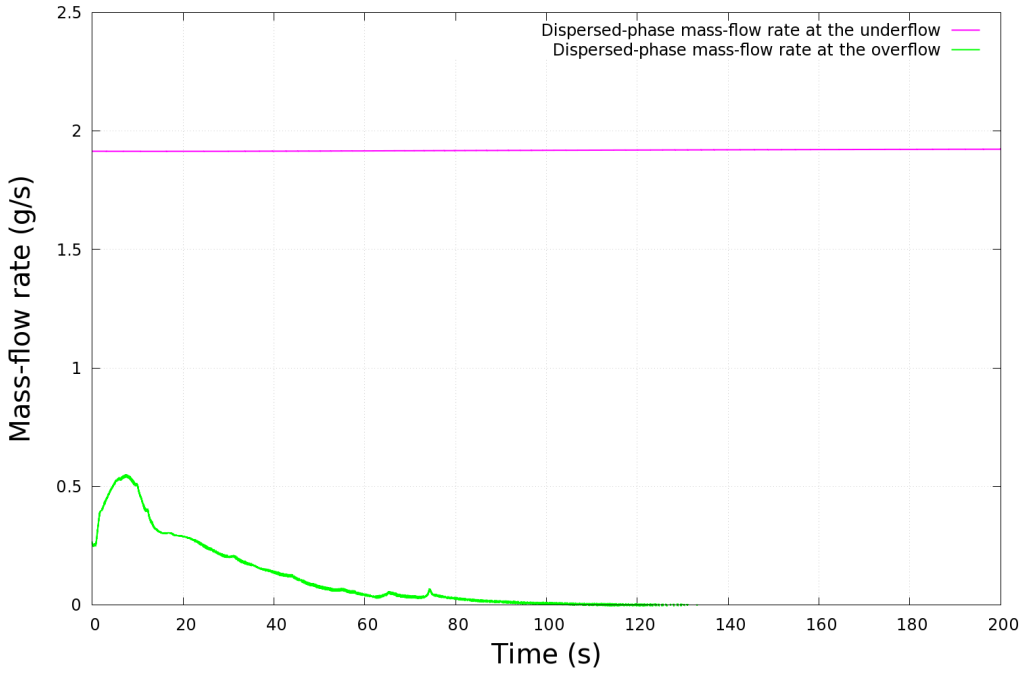


Figure 5.18: Time series plot of dispersed-phase mass-flow rates at outlets on optimal design for case 2.

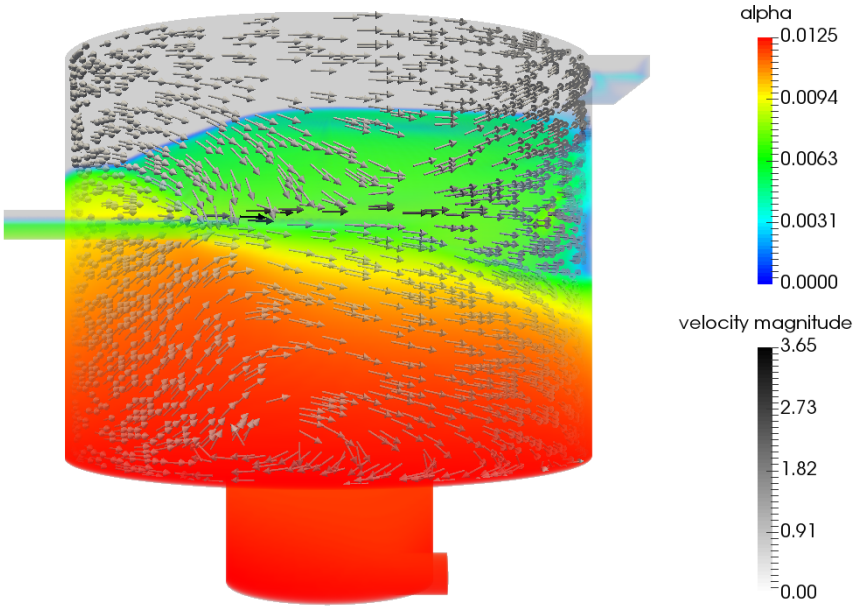


Figure 5.19: Depiction of α and velocity fields after 200 seconds on optimal design for case 2.

Case 3

In this case, the drift flux equations were solved on the initial design for a simulation run of 1,800 seconds, using the initial and boundary conditions for case 1 with the exception of the hydraulic loading rate, which was reduced from 3 l/s/m^2 to 2 l/s/m^2. This corresponds to an inlet velocity of 0.92 m/s and, to simulate a pump extracting sludge at one tenth the inlet flow rate, the underflow velocity was set to 0.04 m/s.

Figure 5.20 shows a time series plot of the dispersed-phase mass-flow rates at the outlets. Its main characteristics are very similar to those seen in cases 1, the only differences being specific values due to the reduced hydraulic loading rate used. In this case, the dispersed-phase mass-flow rate increased sharply between 920 and 930 seconds, confirmed in Figures 5.21 and 5.22, which show the dispersed-phase increase significantly at the underflow between those times. Figure 5.23 shows the quasi-steady state solution of the α and velocity fields after 1,800 seconds.

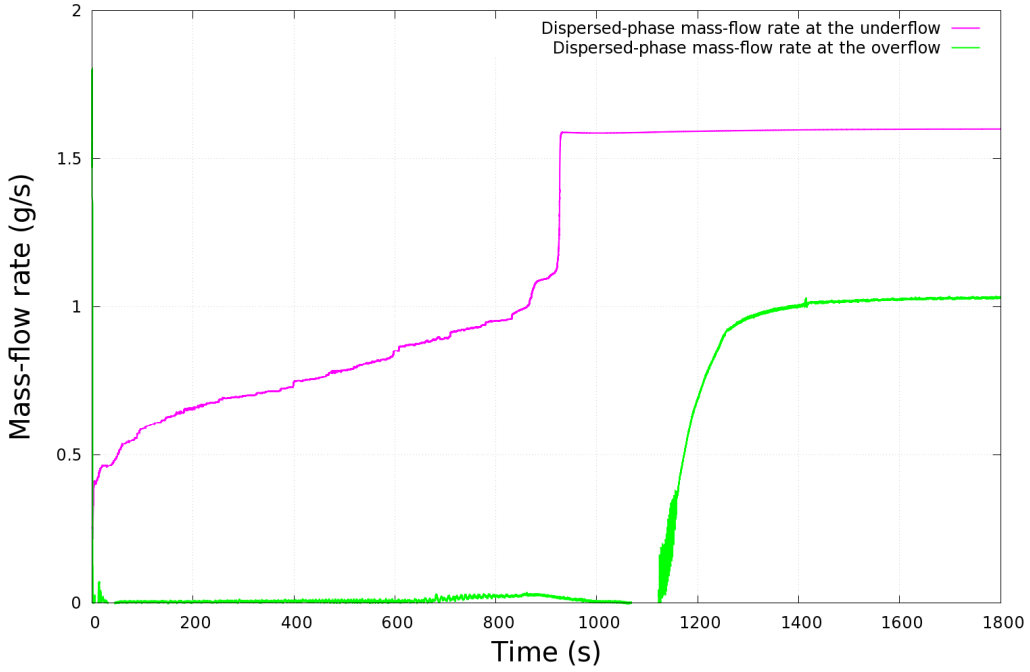


Figure 5.20: Time series plot of dispersed-phase mass-flow rates at outlets on initial design for case 3.

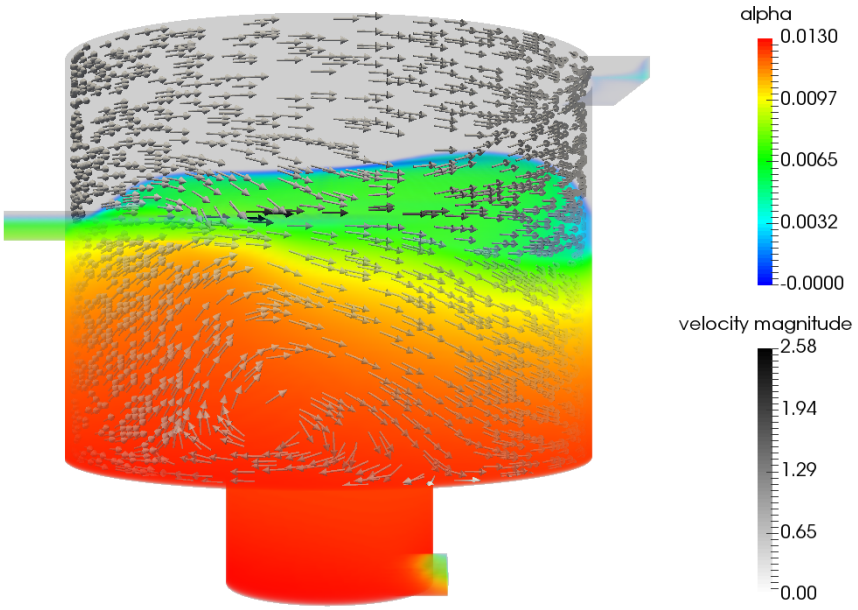


Figure 5.21: Depiction of α and velocity fields after 920 seconds on initial design for case 3.

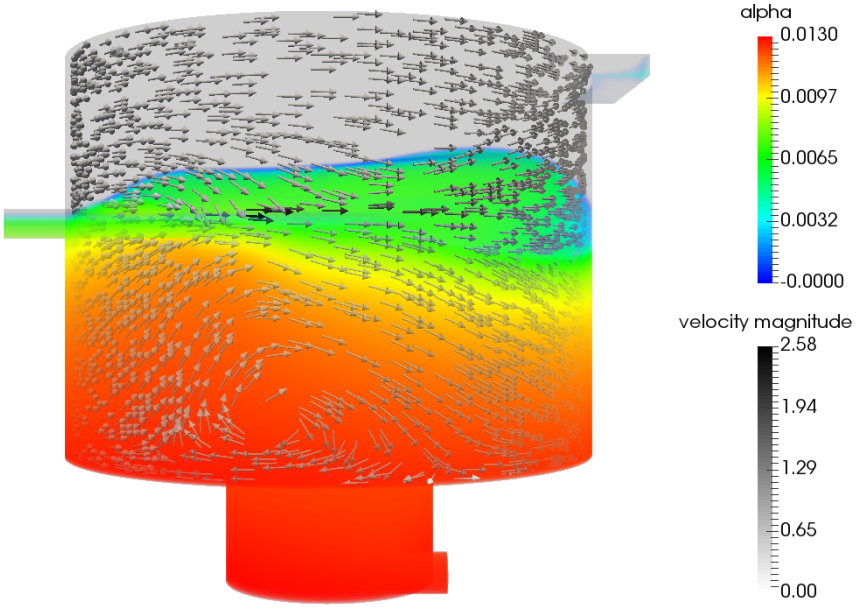


Figure 5.22: Depiction of α and velocity fields after 930 seconds on initial design for case 3.

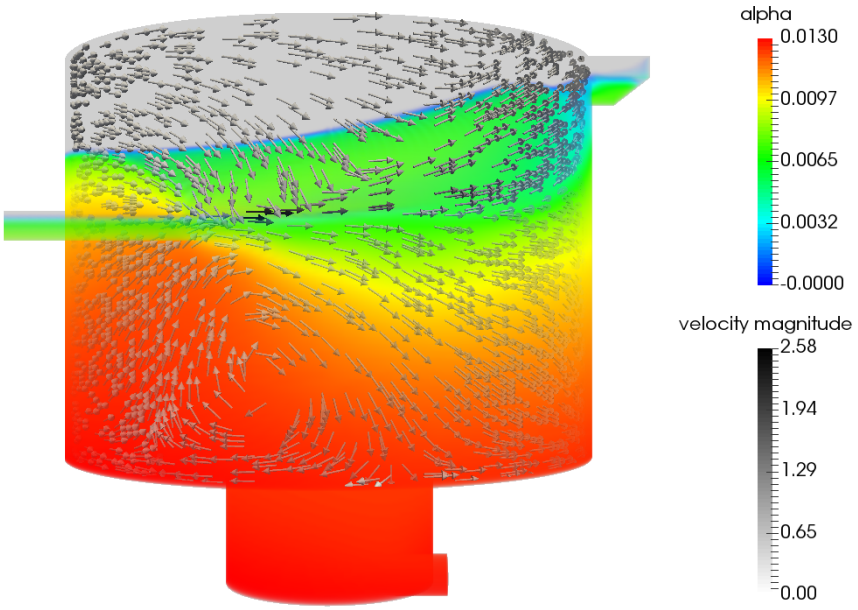


Figure 5.23: Depiction of α and velocity fields after 1,800 seconds on initial design for case 3.

The drift flux equations were then solved on the optimal design for a simulation run of 100 seconds, using the quasi-steady state solution on the initial design as the initial conditions. Figure 5.24 shows the dispersed-phase mass-flow rate at the outlets and it is noticeable that the mass-flow rate at the overflow increased during the simulation. Figure 5.25 shows the α and velocity fields after 100 seconds and it can be seen that the dispersed-phase at the overflow has increased compared with the initial design. Using Eqn. (5.2.2), the performance of the initial and optimal designs are 55% and 48%, respectively, Table 5.7, which represents a 13% reduction in performance of the optimal design compared with the initial design.

Table 5.7: Performance and dispersed-phase mass-flow rates at inlet and outlets of initial and optimal designs for case 3.

	Initial	Optimal
Dispersed-phase mass-flow rate at inlet (g/s)	2.27	2.27
Dispersed-phase mass-flow rate at underflow (g/s)	1.60	1.60
Dispersed-phase mass-flow rate at overflow (g/s)	1.03	1.19
Performance	55%	48%

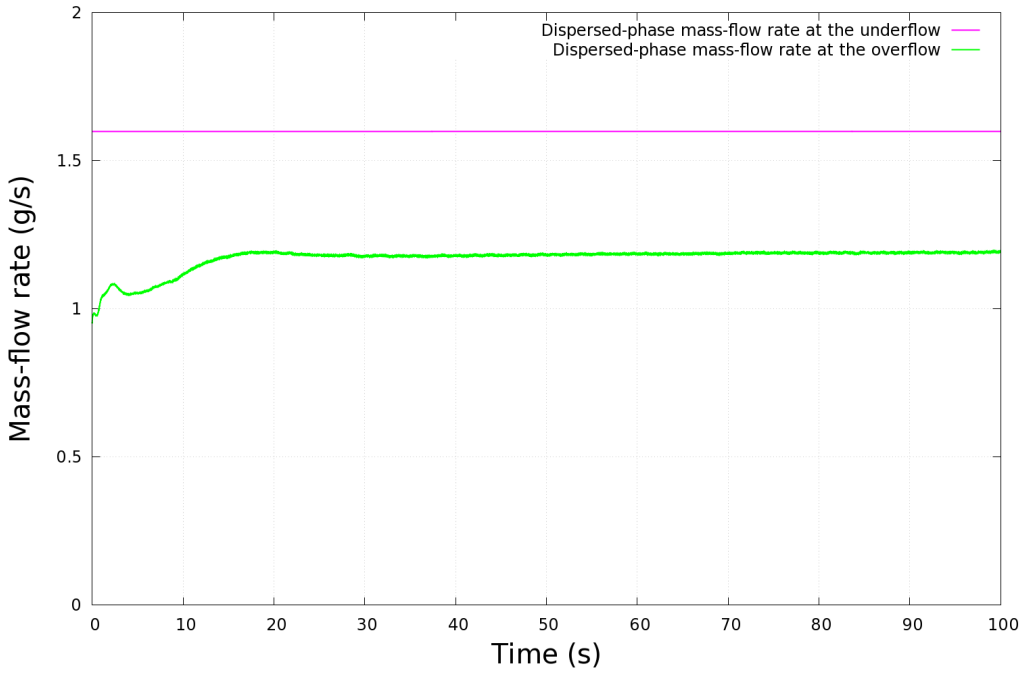


Figure 5.24: Time series plot of dispersed-phase mass-flow rates at outlets on optimal design for case 3.

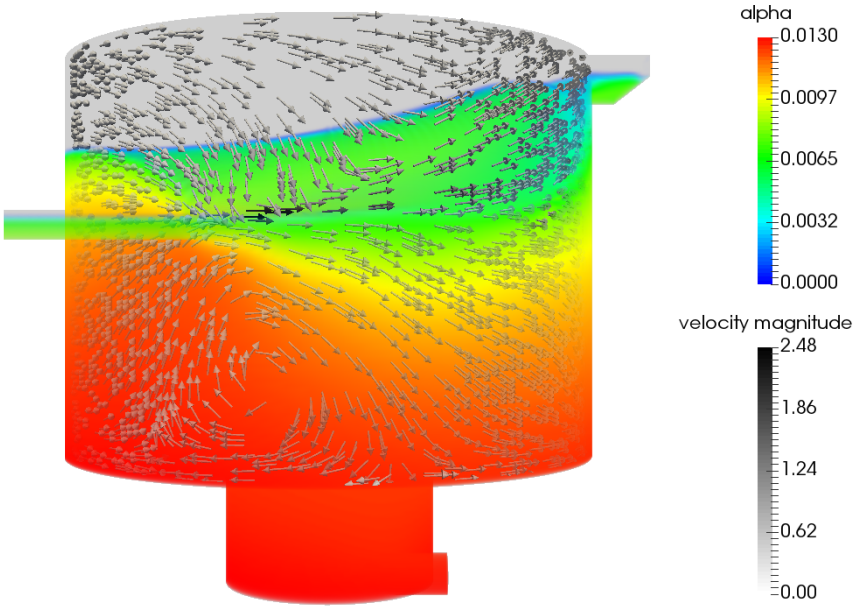


Figure 5.25: Depiction of α and velocity fields after 100 seconds on optimal design for case 3.

5.3.3 Simulation Times

In order to create the optimal design for case 1, the simulations in Table 5.8 were run on 32 processors, except the porosity calculation which only needed to be run on one processor.

Table 5.8: Simulation runs and times to create optimal design for case 1.

Simulation	Runs	Time
Drift flux equations on initial design (1,000 seconds)	1	18 days
Adjoint drift flux equations (50,000 iterations)	1	33 hours
Porosity calculation (1 iteration)	1	33 seconds
Drift flux equations on modified design (100 seconds)	10	12 days

The total time required to create the optimal design for case 1 was one month, Table 5.8, and would have been longer had the drift flux equations on the modified design not been solved from the quasi-steady solutions on the initial design. In general this is only an indication of the total time required. In cases such as 2 and 3, it could take longer because the dispersed-phase would take longer to reach the overflow and, therefore, the initial simulations would take longer to reach a quasi-steady state. These times demonstrate that it can take several weeks to solve CFD simulations and remind us that the adjoint method, Section 2.5.2, is the preferred optimisation technique, because it is independent of the number of design parameters and, therefore, requires fewer CFD calculations than other techniques.

5.4 Summary

Using the methodology in Chapter 4, an optimal design was produced for case 1. The performance of the initial and optimal designs were 45% and 63%, respectively, Table 5.5, representing a 40% improvement of the optimal design compared with the initial design. The capacity of the HDVS was also reduced, because cells that were counterproductive for the flow in terms of the objective function were converted into walls. However, a very small proportion of cells were converted into walls and all were situated next to an existing wall. Consequently, the capacity of the HDVS was reduced by less than 0.03% and manufacture of the optimal design would be relatively straight-forward, because none of the components would need to be suspended inside the domain.

The two designs were then compared on two more test cases: case 2 was identical to case 1 except for the dispersed-phase volume fraction at the inlet, which was reduced from 0.2% to 0.1%, and case 3 was identical to case 1 except for the hydraulic loading rate, which was reduced from 3 to 2 l/s/m^2.

In case 2, the performance of the initial and optimal designs were 86% and 100%, respectively, Table 5.6, representing a 16% improvement of the optimal design compared with the initial design. However in case 3, the performance of the initial and optimal designs were 55% and 48%, respectively, Table 5.7, representing a 13% reduction in performance of the optimal design compared with the initial design. Therefore, the performance of the optimal design would appear to be sensitive to the hydraulic loading rate.

Chapter 6

Discussion

This chapter is divided into two sections: the first discusses the distinctive features of the simulations run in Chapter 5, and the second critically reviews the methodology developed in Chapter 4 and suggests recommendations for its improvement.

6.1 Distinctive Features of the Current Simulations

6.1.1 Optimal Design

The results of Chapter 5 demonstrated that the performance of an HDVS could be significantly improved by applying the methodology of Chapter 4. Specifically, it showed that by fitting the device, illustrated in Figures 5.14 and 5.15, to a wall of the HDVS, the performance of the device could be improved by 40%. Furthermore, the capacity of the HDVS would hardly be reduced (by less than 0.03%) and installation of the device would be relatively straight-forward, because it could be attached to an existing wall.

The function of the device is analogous to that of a car spoiler, which reduces unfavourable air movement along the body of the car to improve its performance. Similarly, the HDVS device reduces the dispersed-phase transported towards the overflow to improve the performance of the HDVS. It achieves this, because it is positioned where the sensitivity of the Lagrange function and the impact on the objective function are greatest, Eqns. (4.1.50) and (4.3.2) respectively. Specifically, cells with a large dispersed-phase volume fraction, α, and high primal and adjoint velocities pointing in the direction of the overflow have the greatest influence on these equations. This can be seen most noticeably in the relationship between the position of the primal velocity vectors pointing towards the overflow and the converted wall cells in the optimal design, Figures 5.4 and 5.15 respectively.

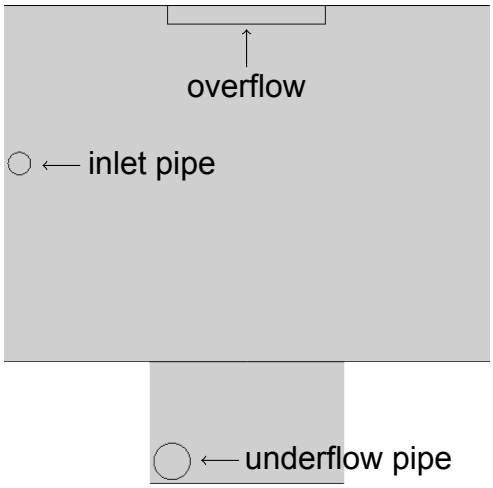
6.1.2 Velocity Fields

As seen in Figures 5.14 and 5.15, the optimal design is created by converting cells into walls in the section between the inlet pipe and overflow, adjacent to the cylinder wall. Therefore, in order to study the effect these converted wall cells have on the velocity field in this region, a cylindrical slice has been taken 10 mm inside the HDVS wall and is viewed from outside the HDVS looking directly towards the overflow. Figures 6.1a and 6.1b show the outline of the initial and optimal designs, respectively, from this perspective.

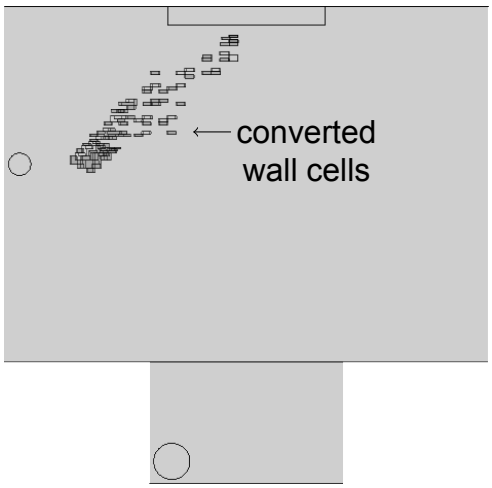
In Figures 6.1c and 6.1d, the velocity magnitude of the mixture is plotted onto the cylindrical slice for the initial and optimal designs, respectively. They show that the contours of the velocity magnitude in the optimal design extend further round the HDVS than in the initial design. The converted wall cells in the optimal design appear to have the effect of accelerating the mixture around the HDVS and underneath the overflow further than in the initial design.

In Figures 6.1e and 6.1f, arrows are superimposed onto Figures 6.1c and 6.1d, respectively, to indicate the direction of the mixture velocity. They confirm that the mixture is being transported further around and away from the overflow, which would have the effect of extending the residence time of the mixture in the HDVS, therefore allowing the dispersed-phase more time to settle, resulting in less dispersed-phase at the overflow.

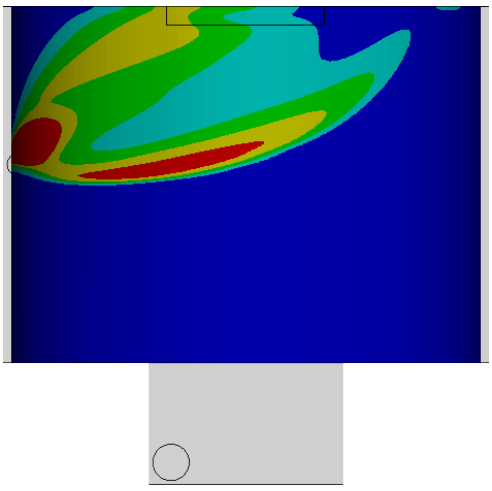
The difference between Figures 6.1e and 6.1f is relatively subtle. However, as explained in Section 6.2.1, the optimal design is only the first iteration of a longer process, and it is anticipated that subsequent iterations would yield a more noticeable difference between the initial and optimal designs, resulting in a more pronounced difference between the flow fields within the two designs.



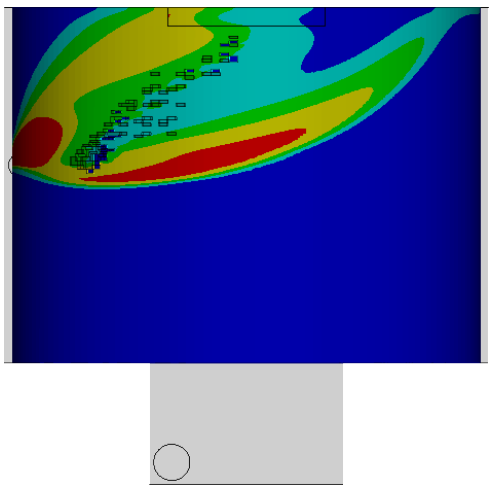
(a) Initial design: outline.



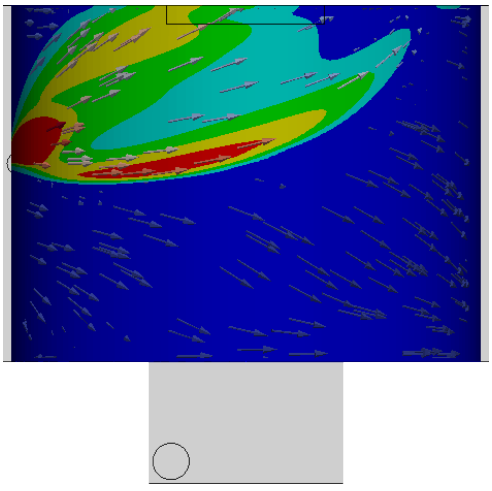
(b) Optimal design: outline.



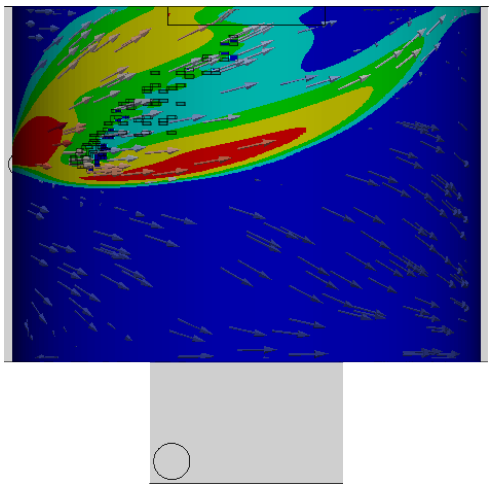
(c) Initial design: velocity magnitude.



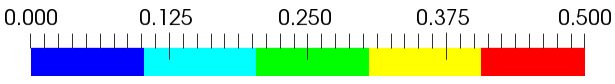
(d) Optimal design: velocity magnitude.



(e) Initial design: velocity vectors.



(f) Optimal design: velocity vectors.



(g) Legend: velocity magnitude.

Figure 6.1: Velocity fields comparison on initial and optimal designs for case 1.

6.1.3 Robustness of the Optimal Design

The performance of the optimal design was also assessed on two more cases. Initial results appear to suggest that the optimal design outperforms the initial design when the dispersed-phase volume fraction at the inlet is changed, but performs worse than the initial design when the hydraulic loading rate is changed. More tests need to be performed before drawing more explicit conclusions. However if this trend were to be confirmed, commercial designs would need to include an optimal hydraulic loading rate in their specifications.

6.1.4 Singularity

In case 1, the drift flux equations were solved on the initial design and, as the simulation progressed, the dispersed-phase mass-flow rate at the underflow increased sharply between 550 and 560 seconds, Figure 5.5, which is confirmed in Figures 5.4e and 5.4f. Similar results were seen in case 3, Figure 5.20, between 920 and 930 seconds, Figures 5.21 and 5.22. During these times, the simulations appear to exhibit catastrophic behaviour, possibly triggered by the build-up of dispersed-phase in the HDVS and consequent pressure at the underflow. It is recommended to perform validation studies of these simulations to verify whether these tipping points occur in practice.

This page is intentionally left blank, to show the procedure on facing pages.

6.2 Critical Review of the Methodology

6.2.1 Improved Optimisation Procedure

In Chapter 5, the methodology of Chapter 4 was applied to a simplified HDVS, in order to find where to modify its internal shape to improve its performance. The optimisation procedure was illustrated with the aid of Figure 4.1b in conjunction with the flow chart in Figure 4.2 to show the path to the optimal design at C.

In this section, the optimisation procedure is improved, using Figure 6.2 in conjunction with the flow chart in Figure 6.3, to show the path to the optimal design at P. This improved procedure consists of an inner- and outer-loop, Figure 6.3. The outer-loop is repeated to climb the path of steepest ascent, Figure 6.2, from O $\rightarrow$ A $\rightarrow$ B $\rightarrow$ C $\rightarrow$ D and so on $\rightarrow$ P. The inner-loop is used to find the optimal design for each outer-loop cycle; it is the same loop used in Figure 4.2 to find the optimal design at C. The difference between the two procedures is that the original procedure has only one outer-loop cycle, ending at C, whereas the improved procedure repeats the outer-loop cycle until it reaches the optimal design at P.

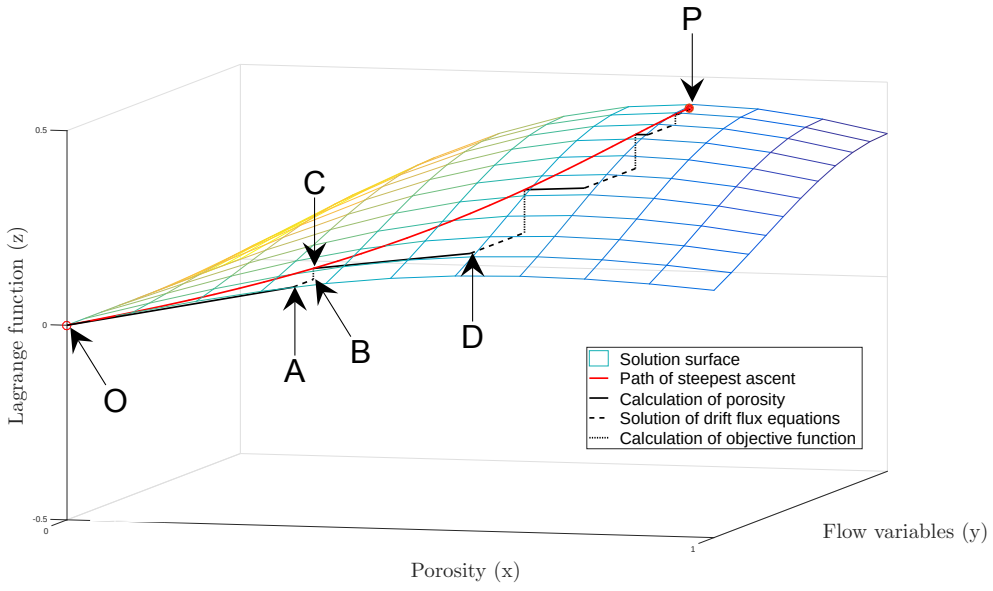


Figure 6.2: Graphical representation of the improved procedure.

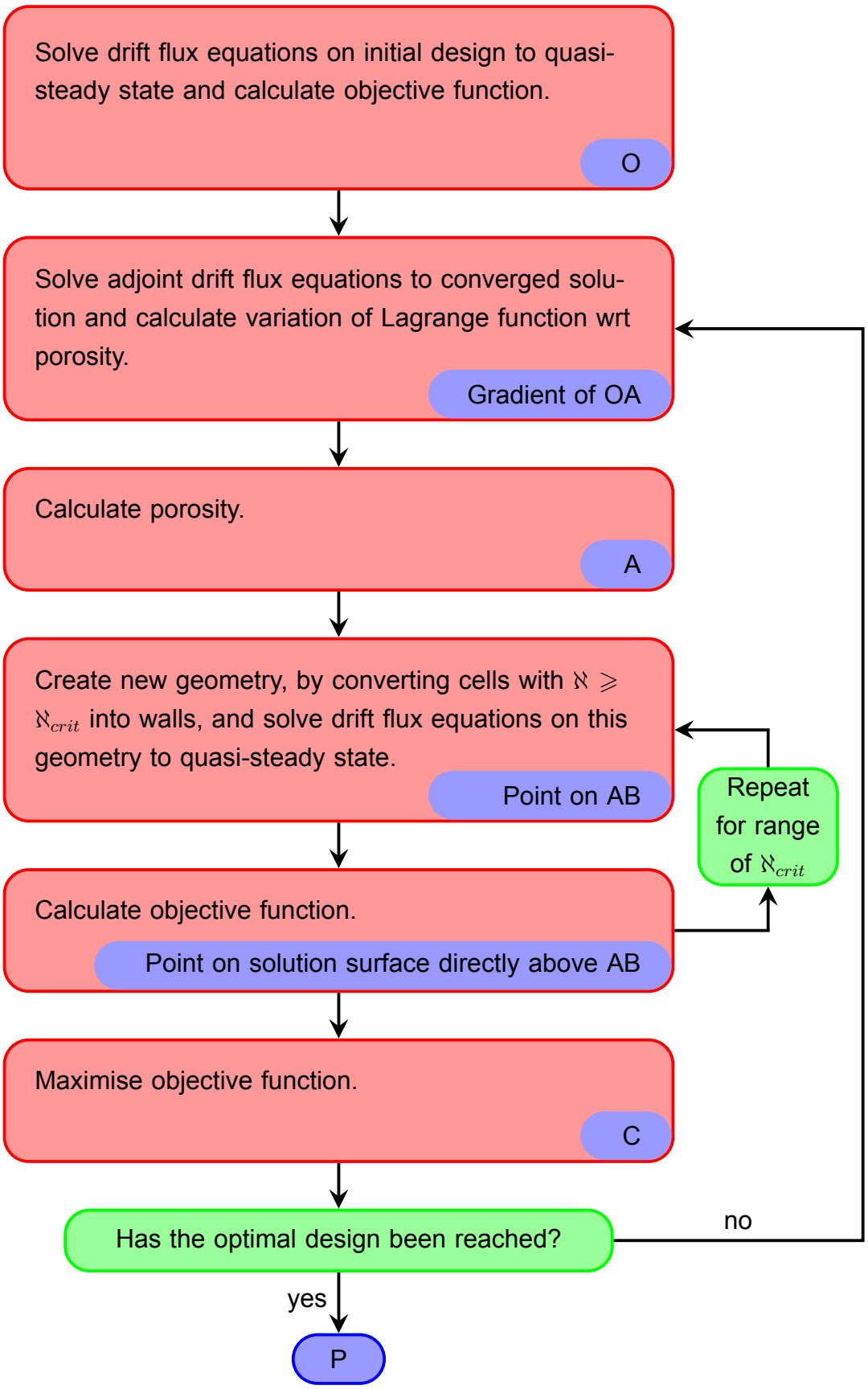


Figure 6.3: Flow chart of the improved procedure. State of system represented by examples in blue boxes, corresponding to symbols in Figure 6.2.

6.2.2 Cells in Confined Spaces

Efforts were made to implement the improved optimisation procedure, Section 6.2.1, but it failed at the start of the second outer-loop cycle, because the adjoint drift flux equations were not solved to a converged solution on the optimal design from the first cycle. Two of the cells that caused the simulation to crash are coloured red in the plan and elevation views of this design, Figures 6.4 and 6.5 respectively. These cells are located in between the outer wall and converted wall cells, coloured dark grey, where the adjoint equations are numerically sensitive to the wall boundary condition for the adjoint variable, β. Therefore, in order to solve the adjoint equations to a converged solution, it would be necessary to identify which cells are in confined spaces and to convert them into walls. It is recommended to develop an algorithm to do this, and to include it in the red box at the start of the outer-loop, Figure 6.3.

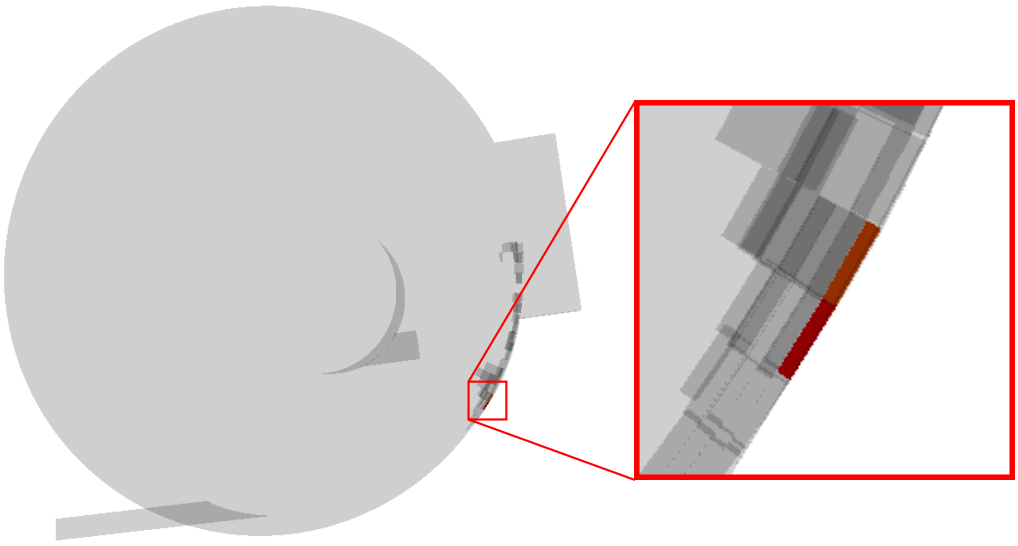


Figure 6.4: Plan view of the optimal design from the first cycle, showing two red cells that caused the simulation of the adjoint equations to crash.

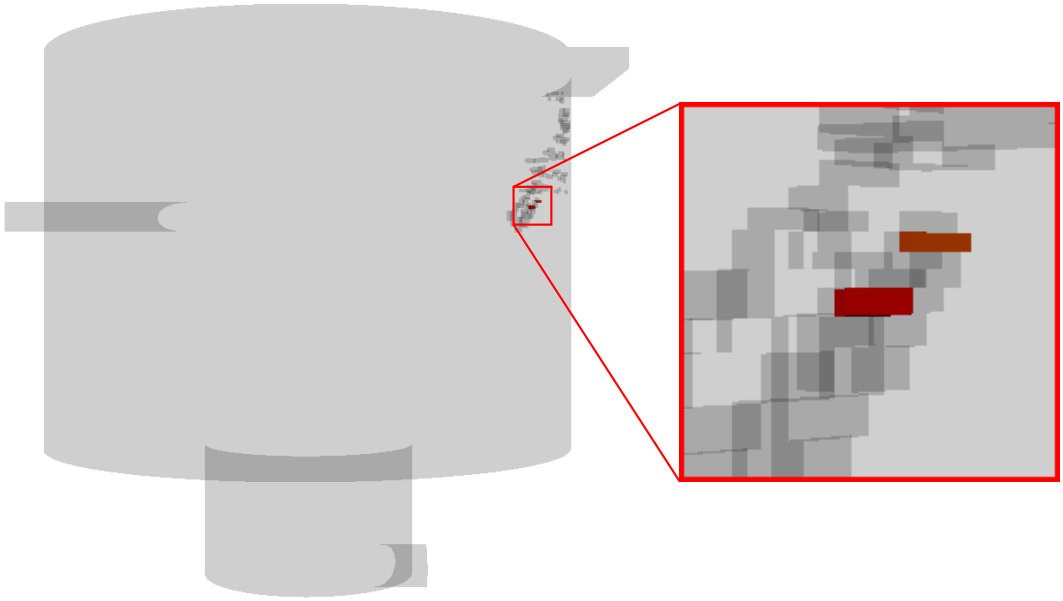


Figure 6.5: Elevation view of the optimal design from the first cycle, showing two red cells that caused the simulation of the adjoint equations to crash.

6.2.3 Geometry Optimisation

The main limitation of topology optimisation is that the initial design can only be modified by removing cells from the mesh. Veerapen et al. (2005) reported that the capture of solids could be improved by moving the overflow away from the centre of the cylinder and by increasing its area. These modifications could not be discovered using topology optimisation and it is recommended to use geometry optimisation to supplement the results obtained from topology optimisation.

6.2.4 Global Search

The main limitation of the adjoint method is that it only offers a local search of the solution space, because it is a gradient-based method. In order to approach a global search, it would be necessary to start the search in other parts of the solution space, using different initial designs, such as an elliptical cylinder or an existing commercial design. When the optimal designs from all these searches have been found, their performance could be compared to find the overall optimal design.

Chapter 7

Conclusions and Further Work

7.1 Conclusions

The main conclusions of this work are as follows:

1. A validation study was performed to assess whether OpenFOAM® can be used to reliably model the flow of water in a scale model HDVS. The results demonstrate that the correlation between experimental and simulation results depends on the mesh structure and turbulence model used. The simulation which produced the best fit to the experimental measurements used the LRR turbulence model and was run on a mesh, which had the appropriate boundary layer distance to use with wall functions. The simulations which used the eddy viscosity models or no turbulence model produced results that consistently overestimated the experimental velocities, though the latter took significantly less time to converge.
2. A methodology was devised to optimise the performance of HDVS through modification of its design, by deriving the adjoint drift flux equations with Darcy porosity term and computing the topological sensitivity at each cell. Cells with porosity above a critical value are removed from the domain to find the design which optimises the removal of solids from wastewater.
3. This methodology was applied to a simplified HDVS, containing none of the internal components present in the validation study, to establish whether it could be used in practice. Boundary conditions were used, corresponding to typical values used in practice, and an optimal HDVS design was found. The performance of the initial and optimal designs were 45% and 63%, respectively, representing a 40% improvement of the optimal design compared with the initial design. These designs were then subjected to a different hydraulic loading rate and different dispersed-phase volume fraction at the inlet, to assess the performance variation in these circumstances. With half the solids' concentration at the inlet, the optimal design removed

all of the solids from the wastewater. However, when it was subjected to a lower hydraulic loading rate, it removed fewer solids compared with the initial design.

4. Initial results suggest that the optimal design is sensitive to hydraulic loading rate, but further tests need to be performed before drawing more explicit conclusions.

7.2 Implications

The methodology developed in Chapter 4 has been shown to deliver positive results (Chapter 5), which have been explained using CFD to analyse the flow within the optimal design (Chapter 6). Previously, in order to improve the performance of an HDVS, modifications to the HDVS were made using engineering judgement. However, there was no procedure to help the engineer make those decisions. With the aid of the methodology developed in Chapter 4, there now exists the potential to use CFD to modify the shape of an HDVS to optimise its performance.

However, the implications of this research are not limited to HDVS nor to wastewater treatment. The methodology could be applied to any device that separates solids from liquid or even two immiscible liquids, for example potato starch from water or milk into cream and skimmed milk. Before this work can be extended, the following measures are recommended to take this research from a proof of concept to a fully developed and validated approach.

7.3 Further Work

The following measures are recommended; the first three relate to validation studies and the following four to modifications in the optimisation procedure.

1. Extend the validation study to multiphase flow on the initial and optimal designs for the simplified HDVS as described in the results chapter, employing as the dispersed-phase: *a*) grit and *b*) a synthetic substitute for sludge.
2. Create a smooth surface for the optimal design, where cells are removed from the domain, and solve the drift flux equations on this geometry to verify that the results are still valid.
3. Apply the optimisation procedure to the design incorporating internal components as used in the validation chapter, to obtain an optimal design for it. Subsequently, carry out the following actions on the initial and optimised designs:

   (a) extend the validation study to multiphase flow and

(b) modify the hydraulic loading rate and solids' concentration at the inlet, to ascertain which design performs better under the new conditions.

4. Update the adjoint equations to include the settling velocity model used by Takacs et al. (1991), Eqns. (2.2.7), which applies to the discrete particle, flocculent and hindered settling stages.
5. Update the adjoint equations to include the local viscosity arising due to variations in the dispersed-phase volume fraction, α, throughout an HDVS.
6. Create an algorithm to remove confined cells from the domain, Section 6.2.2, in order to implement the improved optimisation procedure, Section 6.2.1.
7. Include the second term of the adjoint strain tensor, $(\nabla \mathbf{u})^T$, which was omitted from the adjoint momentum equation to facilitate convergence of the solver.

Publications

The following work from this thesis has been presented at conferences:

1. Results from the validation study were presented at the *Advances in Numerical Modelling of Hydrodynamics Workshop* in the University of Sheffield, (Grossberg et al., 2015).
2. The optimisation methodology was presented at the *European Congress on Computational Methods in Applied Sciences and Engineering (ECCOMAS)* in Greece, 2016.

Appendices

Appendix A

Validation Study Outputs

A.1 Calibration Charts

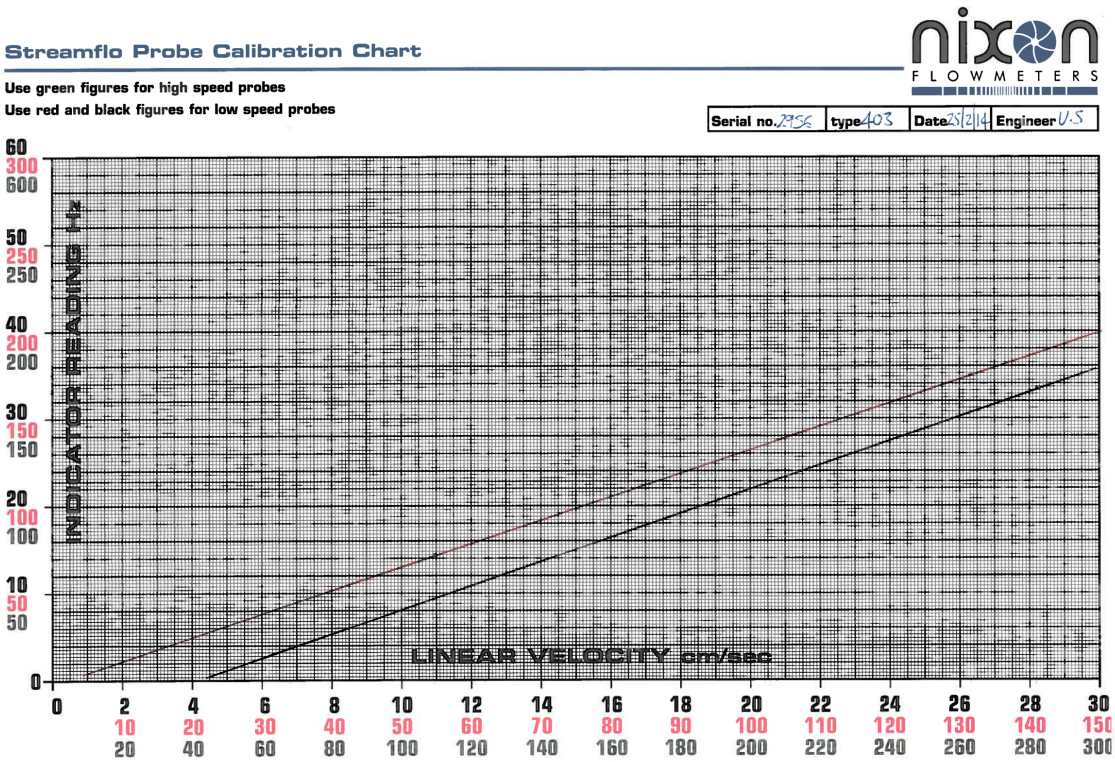


Figure A.1: Nixon Flow Meters' Streamflo Probe Calibration Chart.

Equations of black and red lines, respectively,

$$v = 0.722f + 4.156 \tag{A.1.1}$$

$$v = 0.743f + 1.767, \tag{A.1.2}$$

where f denotes frequency and v denotes velocity, and equation of cubic spline interpolation between them,

$$v = 0.0003f^3 - 0.0192f^2 + 1.0166f + 2.8317. \tag{A.1.3}$$

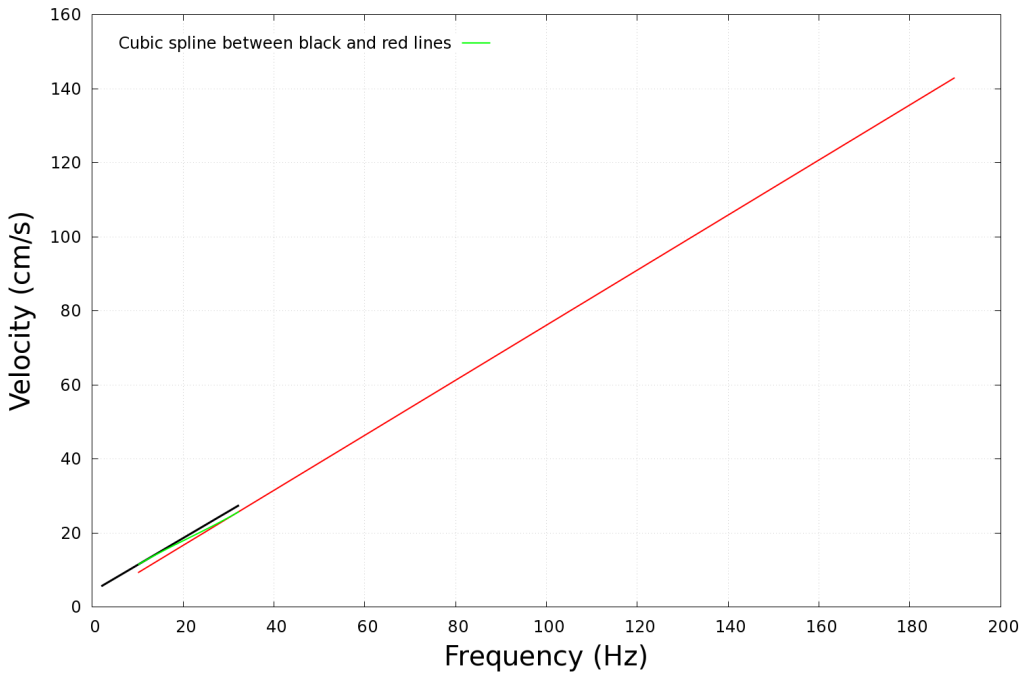


Figure A.2: Calibration chart to include cubic spline interpolation between black and red lines.

A.2 Frequency Readings

All frequency readings (Hz) are for an hydraulic loading rate of 3 l/s/m^2. The azimuthal angle, radial distance and height are specified for each reading, which is taken five times, represented by: R1, R2, R3, R4 and R5.

Azimuthal angle = 22.5°, radial distance = 0.01 m.

Height (m)	R1	R2	R3	R4	R5
0.18	32.5	32.4	32.7	33.0	32.5
0.15	35.1	35.9	36.1	35.9	35.6
0.12	37.5	37.2	37.4	36.9	37.8
0.09	36.2	36.2	36.6	36.5	35.8
0.06	38.3	38.8	38.2	37.6	38.4
0.03	34.4	34.3	33.7	33.2	34.0
0	135	137	138	141	143
-0.03	77.8	76.9	74.8	74.0	75.2
-0.06	38.8	39.0	39.1	39.2	38.7
-0.09	40.7	40.8	41.4	41.4	40.8

Azimuthal angle = 22.5°, radial distance = 0.04 m.

Height (m)	R1	R2	R3	R4	R5
0.18	36.8	36.7	37.1	36.8	36.9
0.15	38.9	38.4	38.6	38.4	39.2
0.12	38.9	38.4	39.0	38.8	38.5
0.09	35.1	35.3	35.0	34.9	34.9
0.06	28.8	29.3	28.3	28.8	29.3
0.03	25.6	26.1	27.1	25.9	27.0
0	26.5	27.1	27.5	27.2	27.6
-0.03	30.1	30.0	30.5	30.2	30.6
-0.06	30.3	30.4	30.8	30.9	30.4
-0.09	30.3	29.4	29.5	29.5	29.4

Azimuthal angle = 22.5°, radial distance = 0.07 m.

Height (m)	R1	R2	R3	R4	R5
0.18	34.7	34.6	34.7	35.0	34.8
0.15	35.2	35.6	35.0	35.1	35.3
0.12	33.3	33.7	33.7	33.5	33.6
0.09	29.2	28.9	29.6	29.0	29.2
0.06	23.2	22.7	22.8	23.9	23.1
0.03	19.0	19.7	19.1	19.7	19.2
0	19.5	20.1	19.9	19.8	20.1
-0.03	21.7	21.1	21.2	21.6	21.2
-0.06	22.1	22.3	22.6	22.3	22.8
-0.09	21.4	21.6	21.9	21.3	21.7

Azimuthal angle = 90°, radial distance = 0.01 m.

Height (m)	R1	R2	R3	R4	R5
0.18	32.1	32.8	33.6	32.2	32.7
0.15	34.8	34.6	34.8	35.0	34.9
0.12	52.9	52.6	53.1	51.5	52.5
0.09	44.4	44.8	44.6	44.9	44.7
0.06	48.6	48.4	47.4	48.2	48.4
0.03	48.5	48.8	49.1	49.9	48.9
0	43.0	44.5	42.8	43.3	44.0
-0.03	39.8	40.6	40.7	40.6	42.1
-0.06	37.8	38.0	38.3	37.6	37.3
-0.09	46.4	45.6	44.9	45.3	45.6

Azimuthal angle = 90°, radial distance = 0.04 m.

Height (m)	R1	R2	R3	R4	R5
0.18	35.5	35.9	35.9	35.7	35.4
0.15	38.1	37.4	37.9	37.9	37.5
0.12	37.4	36.9	37.1	37.1	37.1
0.09	35.8	35.7	35.1	35.6	35.7
0.06	31.7	31.9	31.3	31.8	32.1
0.03	26.1	26.3	25.9	26.2	26.3
0	22.2	22.2	22.3	22.4	22.5
-0.03	20.9	21.4	20.4	20.5	21.0
-0.06	22.7	22.0	21.6	21.9	22.4
-0.09	25.3	24.9	24.9	24.7	24.9

Azimuthal angle = 90°, radial distance = 0.07 m.

Height (m)	R1	R2	R3	R4	R5
0.18	32.9	33.3	32.8	32.8	33.2
0.15	29.7	30.3	30.3	30.1	30.3
0.12	29.7	28.8	29.3	29.2	28.8
0.09	27.8	28.1	28.3	28.2	28.4
0.06	25.1	25.5	25.6	25.3	24.9
0.03	22.0	22.3	21.9	21.6	21.8
0	19.7	19.2	19.3	19.0	18.8
-0.03	17.5	17.5	17.3	17.6	17.5
-0.06	16.6	16.7	16.7	16.4	17.0
-0.09	16.8	16.7	17.0	17.0	16.8

Azimuthal angle = 180°, radial distance = 0.01 m.

Height (m)	R1	R2	R3	R4	R5
0.18	27.4	26.7	27.8	26.6	27.7
0.15	34.6	35.7	35.9	34.5	35.0
0.12	39.6	39.3	39.8	39.3	39.2
0.09	44.3	44.1	43.9	44.1	44.4
0.06	47.0	46.7	46.2	46.6	46.6
0.03	46.1	46.1	47.2	46.5	47.3
0	44.0	43.6	44.2	44.5	43.5
-0.03	46.0	46.3	46.1	46.5	46.2
-0.06	44.1	42.3	43.2	43.1	43.7
-0.09	42.7	42.1	42.6	42.5	43.1

Azimuthal angle = 180°, radial distance = 0.04 m.

Height (m)	R1	R2	R3	R4	R5
0.18	19.8	20.9	20.3	20.0	20.1
0.15	25.3	25.1	24.8	25.1	24.8
0.12	28.1	28.0	27.8	27.8	27.7
0.09	27.9	27.7	27.6	27.7	27.6
0.06	27.4	28.0	27.8	27.9	27.8
0.03	27.9	27.9	28.0	27.5	28.0
0	28.0	28.5	28.0	28.8	28.4
-0.03	29.2	29.0	28.7	29.1	28.6
-0.06	30.1	30.6	29.6	30.3	31.0
-0.09	32.6	33.2	33.1	32.6	32.6

Azimuthal angle = 180°, radial distance = 0.07 m.

Height (m)	R1	R2	R3	R4	R5
0.18	7.0	6.7	6.6	5.8	6.9
0.15	15.8	14.3	14.7	15.6	14.4
0.12	16.5	18.5	18.0	17.4	17.7
0.09	21.5	21.1	21.5	20.5	21.0
0.06	22.7	22.6	22.6	22.0	22.4
0.03	23.1	23.4	23.4	23.3	23.6
0	23.7	24.3	24.2	23.9	24.0
-0.03	25.1	25.1	25.6	25.2	24.4
-0.06	24.5	24.3	24.5	24.8	24.5
-0.09	21.8	21.5	22.2	22.0	21.1

Azimuthal angle = 270°, radial distance = 0.01 m.

Height (m)	R1	R2	R3	R4	R5
0.18	22.0	24.6	23.0	23.3	23.8
0.15	32.9	32.4	32.4	33.1	32.6
0.12	32.3	32.4	32.0	32.1	31.5
0.09	32.6	32.8	33.5	33.4	33.0
0.06	35.2	35.0	35.4	35.3	35.5
0.03	35.8	36.2	35.3	35.8	36.0
0	37.1	36.6	37.0	36.9	36.3
-0.03	36.4	35.6	35.8	35.8	35.9
-0.06	38.7	38.9	39.1	39.4	39.3
-0.09	38.8	38.7	39.1	38.6	39.2

Azimuthal angle = 270°, radial distance = 0.04 m.

Height (m)	R1	R2	R3	R4	R5
0.18	24.3	24.4	24.5	23.5	24.4
0.15	13.7	13.9	14.7	14.5	14.4
0.12	9.7	9.9	10.6	10.5	9.2
0.09	17.0	19.0	19.7	19.0	19.0
0.06	28.5	28.3	28.6	28.8	29.1
0.03	31.0	30.6	30.7	30.8	31.0
0	31.7	31.5	31.8	32.2	31.8
-0.03	31.2	31.0	30.7	30.8	31.3
-0.06	31.3	32.0	32.0	30.9	30.8
-0.09	28.7	28.5	27.9	27.6	28.2

Azimuthal angle = 270°, radial distance = 0.07 m.

Height (m)	R1	R2	R3	R4	R5
0.18	4.2	5.5	4.3	4.4	4.5
0.15	3.6	1.7	2.9	2.4	2.6
0.12	3.8	3.7	4.3	4.8	3.2
0.09	6.7	6.4	5.2	6.5	6.0
0.06	13.2	13.0	12.5	12.5	12.4
0.03	14.9	14.8	14.9	14.6	14.2
0	16.1	16.3	16.6	16.1	16.4
-0.03	19.5	19.8	19.3	19.4	19.9
-0.06	21.6	20.5	20.5	20.9	20.6
-0.09	20.1	20.3	20.3	20.7	20.0

A.3 Mesh Statistics

A.3.1 Coarse Isotropic Mesh

```
Create time

Create polyMesh for time = 0

Time = 0

Mesh stats
    points:         309504
    faces:          3409634
    internal faces: 3287274
    cells:          1674227
    faces per cell: 4
    boundary patches: 19
    point zones:    0
    face zones:     0
    cell zones:     0

Overall number of cells of each type:
    hexahedra:   0
    prisms:      0
    wedges:      0
    pyramids:    0
    tet wedges:  0
    tetrahedra:  1674227
    polyhedra:   0

Checking topology...
    Boundary definition OK.
    Cell to face addressing OK.
    Point usage OK.
    Upper triangular ordering OK.
    Face vertices OK.
    Number of regions: 1 (OK).

Checking patch topology for multiply connected surfaces...
    Patch            Faces   Points  Surface topology
    free             7939    4255    ok (non-closed singly connected)
    inlet            434     234     ok (non-closed singly connected)
    overflow         190     119     ok (non-closed singly connected)
```

```
    stand             153     86      ok (non-closed singly connected)
    underflow         57      37      ok (non-closed singly connected)
    valve             22      17      ok (non-closed singly connected)
    wall-baffle       5942    3000    ok (non-closed singly connected)
    wall-central      4236    2203    ok (non-closed singly connected)
    wall-chamber      36462   18495   ok (non-closed singly connected)
    wall-cone         7660    3858    ok (non-closed singly connected)
    wall-deflector    11577   5883    ok (non-closed singly connected)
    wall-dip          12981   6743    ok (non-closed singly connected)
    wall-inlet-frustum 3448   1750    ok (non-closed singly connected)
    wall-inlet-large  8171    4129    ok (non-closed singly connected)
    wall-inlet-small  8906    4522    ok (non-closed singly connected)
    wall-overflow     3765    1993    ok (non-closed singly connected)
    wall-stand        4790    2412    ok (non-closed singly connected)
    wall-tray         4931    2598    ok (non-closed singly connected)
    wall-underflow    696     370     ok (non-closed singly connected)

Checking geometry...
    Overall domain bounding box (-2.21 -0.335 -3.68988e-07) (0.3 0.464923
        1.45)
    Mesh (non-empty, non-wedge) directions (1 1 1)
    Mesh (non-empty) directions (1 1 1)
    Boundary openness (2.40974e-18 5.31536e-16 1.21968e-15) OK.
    Max cell openness = 5.70994e-16 OK.
    Max aspect ratio = 33.745 OK.
    Minimum face area = 7.82196e-08. Maximum face area = 0.000324669. Face
         area magnitudes OK.
    Min volume = 1.14612e-11. Max volume = 1.90189e-06. Total volume =
        0.126897. Cell volumes OK.
    Mesh non-orthogonality Max: 81.8398 average: 27.1911
   *Number of severely non-orthogonal faces: 5222.
    Non-orthogonality check OK.
  <<Writing 5222 non-orthogonal faces to set nonOrthoFaces
    Face pyramids OK.
    Max skewness = 1.85683 OK.
    Coupled point location match (average 0) OK.

Mesh OK.

End
```

A.3.2 Fine Isotropic Mesh

```
Create time

Create polyMesh for time = 0

Time = 0

Mesh stats
    points:         1463025
    faces:          16492536
    internal faces: 16048520
    cells:          8135264
    faces per cell: 4
    boundary patches: 19
    point zones:    0
    face zones:     0
    cell zones:     0

Overall number of cells of each type:
    hexahedra:   0
    prisms:      0
    wedges:      0
    pyramids:    0
    tet wedges:  0
    tetrahedra:  8135264
    polyhedra:   0

Checking topology...
    Boundary definition OK.
    Cell to face addressing OK.
    Point usage OK.
    Upper triangular ordering OK.
    Face vertices OK.
    Number of regions: 1 (OK).

Checking patch topology for multiply connected surfaces...
    Patch             Faces   Points  Surface topology
    free              23169   12148   ok (non-closed singly connected)
    inlet             1324    695     ok (non-closed singly connected)
    overflow          587     335     ok (non-closed singly connected)
    stand             478     257     ok (non-closed singly connected)
    underflow         236     134     ok (non-closed singly connected)
```

```
    valve             106     64      ok (non-closed singly connected)
    wall-baffle       22004   11060   ok (non-closed singly connected)
    wall-central      16624   8482    ok (non-closed singly connected)
    wall-chamber      138972  70016   ok (non-closed singly connected)
    wall-cone         28750   14431   ok (non-closed singly connected)
    wall-deflector    45582   22979   ok (non-closed singly connected)
    wall-dip          48996   25002   ok (non-closed singly connected)
    wall-inlet-frustum 13070 6587     ok (non-closed singly connected)
    wall-inlet-large  31244   15711   ok (non-closed singly connected)
    wall-inlet-small  33392   16834   ok (non-closed singly connected)
    wall-overflow     14566   7502    ok (non-closed singly connected)
    wall-stand        4824    2446    ok (non-closed singly connected)
    wall-tray         19352   9941    ok (non-closed singly connected)
    wall-underflow    740     414     ok (non-closed singly connected)

Checking geometry...
    Overall domain bounding box (-2.21 -0.335 -5.12887e-07) (0.3 0.464923
        1.45)
    Mesh (non-empty, non-wedge) directions (1 1 1)
    Mesh (non-empty) directions (1 1 1)
    Boundary openness (5.0092e-17 -4.20184e-16 3.04748e-15) OK.
    Max cell openness = 4.3753e-16 OK.
    Max aspect ratio = 41.5391 OK.
    Minimum face area = 1.73466e-08. Maximum face area = 0.000141989. Face
         area magnitudes OK.
    Min volume = 1.13408e-12. Max volume = 5.28671e-07. Total volume =
        0.126944. Cell volumes OK.
    Mesh non-orthogonality Max: 84.2083 average: 25.6212
   *Number of severely non-orthogonal faces: 36626.
    Non-orthogonality check OK.
  <<Writing 36626 non-orthogonal faces to set nonOrthoFaces
    Face pyramids OK.
    Max skewness = 2.06657 OK.
    Coupled point location match (average 0) OK.

Mesh OK.

End
```

A.3.3 Coarse Isotropic Mesh with T-Rex® Boundary Layer

```
Create time

Create polyMesh for time = 0

Time = 0

Mesh stats
    points:         309420
    faces:          2131565
    internal faces: 2010829
    cells:          939886
    faces per cell: 4.40734
    boundary patches: 19
    point zones:    0
    face zones:     0
    cell zones:     0

Overall number of cells of each type:
    hexahedra:   0
    prisms:      351012
    wedges:      0
    pyramids:    31838
    tet wedges:  0
    tetrahedra:  557036
    polyhedra:   0

Checking topology...
    Boundary definition OK.
    Cell to face addressing OK.
    Point usage OK.
    Upper triangular ordering OK.
    Face vertices OK.
    Number of regions: 1 (OK).

Checking patch topology for multiply connected surfaces...
    Patch            Faces   Points  Surface topology
    free             6594    4235    ok (non-closed singly connected)
    inlet            261     234     ok (non-closed singly connected)
    overflow         152     119     ok (non-closed singly connected)
    stand            85      86      ok (non-closed singly connected)
    underflow        57      37      ok (non-closed singly connected)
```

```
    valve             22      17      ok (non-closed singly connected)
    wall-baffle       5942    3000    ok (non-closed singly connected)
    wall-central      4236    2203    ok (non-closed singly connected)
    wall-chamber      36462   18495   ok (non-closed singly connected)
    wall-cone         7660    3858    ok (non-closed singly connected)
    wall-deflector    11577   5883    ok (non-closed singly connected)
    wall-dip          12981   6743    ok (non-closed singly connected)
    wall-inlet-frustum 3448   1750    ok (non-closed singly connected)
    wall-inlet-large  8171    4129    ok (non-closed singly connected)
    wall-inlet-small  8906    4522    ok (non-closed singly connected)
    wall-overflow     3765    1993    ok (non-closed singly connected)
    wall-stand        4790    2412    ok (non-closed singly connected)
    wall-tray         4931    2598    ok (non-closed singly connected)
    wall-underflow    696     370     ok (non-closed singly connected)

Checking geometry...
    Overall domain bounding box (-2.21 -0.335 -3.68988e-07) (0.3 0.464923
        1.45)
    Mesh (non-empty, non-wedge) directions (1 1 1)
    Mesh (non-empty) directions (1 1 1)
    Boundary openness (2.40974e-18 5.31536e-16 7.87981e-16) OK.
    Max cell openness = 4.71118e-16 OK.
    Max aspect ratio = 27.1097 OK.
    Minimum face area = 8.08947e-08. Maximum face area = 0.000328603. Face
         area magnitudes OK.
    Min volume = 1.16555e-11. Max volume = 1.8098e-06. Total volume =
        0.126897. Cell volumes OK.
    Mesh non-orthogonality Max: 81.5264 average: 19.7785
   *Number of severely non-orthogonal faces: 622.
    Non-orthogonality check OK.
  <<Writing 622 non-orthogonal faces to set nonOrthoFaces
    Face pyramids OK.
    Max skewness = 2.07414 OK.
    Coupled point location match (average 0) OK.

Mesh OK.

End
```

A.3.4 Fine Isotropic Mesh with T-Rex® Boundary Layer

```
Create time

Create polyMesh for time = 0

Time = 0

Mesh stats
    points:         1314781
    faces:          11547634
    internal faces: 11109006
    cells:          5430675
    faces per cell: 4.17197
    boundary patches: 19
    point zones:    0
    face zones:     0
    cell zones:     0

Overall number of cells of each type:
    hexahedra:   0
    prisms:      881891
    wedges:      0
    pyramids:    52049
    tet wedges:  0
    tetrahedra:  4496735
    polyhedra:   0

Checking topology...
    Boundary definition OK.
    Cell to face addressing OK.
    Point usage OK.
    Upper triangular ordering OK.
    Face vertices OK.
    Number of regions: 1 (OK).

Checking patch topology for multiply connected surfaces...
    Patch             Faces   Points  Surface topology
    free              18748   10468   ok (non-closed singly connected)
    inlet             881     538     ok (non-closed singly connected)
    overflow          349     232     ok (non-closed singly connected)
    stand             192     169     ok (non-closed singly connected)
    underflow         236     134     ok (non-closed singly connected)
```

```
    valve             106     64      ok (non-closed singly connected)
    wall-baffle       22004   11060   ok (non-closed singly connected)
    wall-central      16624   8482    ok (non-closed singly connected)
    wall-chamber      138972  70016   ok (non-closed singly connected)
    wall-cone         28750   14431   ok (non-closed singly connected)
    wall-deflector    45582   22979   ok (non-closed singly connected)
    wall-dip          48996   25002   ok (non-closed singly connected)
    wall-inlet-frustum 13070 6587     ok (non-closed singly connected)
    wall-inlet-large  31244   15711   ok (non-closed singly connected)
    wall-inlet-small  33392   16834   ok (non-closed singly connected)
    wall-overflow     14566   7502    ok (non-closed singly connected)
    wall-stand        4824    2446    ok (non-closed singly connected)
    wall-tray         19352   9941    ok (non-closed singly connected)
    wall-underflow    740     414     ok (non-closed singly connected)

Checking geometry...
    Overall domain bounding box (-2.21 -0.335 -5.12887e-07) (0.3 0.464923
        1.45)
    Mesh (non-empty, non-wedge) directions (1 1 1)
    Mesh (non-empty) directions (1 1 1)
    Boundary openness (4.14604e-17 -4.20184e-16 3.30643e-15) OK.
    Max cell openness = 4.49675e-16 OK.
    Max aspect ratio = 26.5421 OK.
    Minimum face area = 1.49213e-08. Maximum face area = 0.000142546. Face
         area magnitudes OK.
    Min volume = 1.1028e-12. Max volume = 5.28214e-07. Total volume =
        0.126944. Cell volumes OK.
    Mesh non-orthogonality Max: 85.4182 average: 19.0672
   *Number of severely non-orthogonal faces: 1195.
    Non-orthogonality check OK.
  <<Writing 1195 non-orthogonal faces to set nonOrthoFaces
    Face pyramids OK.
    Max skewness = 2.05129 OK.
    Coupled point location match (average 0) OK.

Mesh OK.

End
```

A.4 yPlus Values

A.4.1 Coarse Isotropic Mesh using Realisable k-epsilon Turbulence Model

```
Create time

Create mesh for time = 6041

Time = 6041
Calculating wall distance

Writing wall distance to field y

Reading field U

Reading/calculating face flux field phi

Selecting incompressible transport model Newtonian
Selecting RAS turbulence model realizableKE
realizableKECoeffs
{
    Cmu           0.09;
    A0            4;
    C2            1.9;
    sigmak        1;
    sigmaEps      1.2;
}

Patch 3 named stand y+ : min: 0.000246049 max: 0.000602435 average:
    0.000453466

Patch 4 named underflow y+ : min: 0.0331092 max: 0.0876875 average:
    0.0605553

Patch 5 named valve y+ : min: 0.741033 max: 0.931311 average: 0.861642

Patch 6 named wall-baffle y+ : min: 0.89866 max: 11.4143 average: 6.72203

Patch 7 named wall-central y+ : min: 0.0473223 max: 5.56401 average:
    1.80348

Patch 8 named wall-chamber y+ : min: 0.158616 max: 46.1366 average:
```

```
    15.8512

Patch 9 named wall-cone y+ : min: 0.267469 max: 9.39083 average: 4.57719

Patch 10 named wall-deflector y+ : min: 0.220873 max: 30.536 average:
    9.49311

Patch 11 named wall-dip y+ : min: 2.14978 max: 54.3625 average: 11.4792

Patch 12 named wall-inlet-frustum y+ : min: 2.37749 max: 15.083 average:
    6.07151

Patch 13 named wall-inlet-large y+ : min: 0.801051 max: 4.98392 average:
    3.65406

Patch 14 named wall-inlet-small y+ : min: 5.98702 max: 11.5139 average:
    7.66811

Patch 15 named wall-overflow y+ : min: 1.64453 max: 66.1754 average:
    18.1965

Patch 16 named wall-stand y+ : min: 0.000199986 max: 3.23885 average:
    0.320704

Patch 17 named wall-tray y+ : min: 1.37962 max: 50.9869 average: 9.98053

Patch 18 named wall-underflow y+ : min: 0.0418488 max: 2.17886 average:
    0.603036

Writing yPlus to field yPlus

End
```

A.4.2 Fine Isotropic Mesh using Realisable k-epsilon Turbulence Model

```
Create time

Create mesh for time = 4088

Time = 4088
Calculating wall distance

Writing wall distance to field y

Reading field U

Reading/calculating face flux field phi

Selecting incompressible transport model Newtonian
Selecting RAS turbulence model realizableKE
realizableKECoeffs
{
    Cmu           0.09;
    A0            4;
    C2            1.9;
    sigmak        1;
    sigmaEps      1.2;
}

Patch 3 named stand y+ : min: 0.00113319 max: 0.00362583 average:
    0.0027603

Patch 4 named underflow y+ : min: 0.00359845 max: 0.00547241 average:
    0.00446234

Patch 5 named valve y+ : min: 0.1716 max: 0.22909 average: 0.199374

Patch 6 named wall-baffle y+ : min: 0.102556 max: 6.58433 average: 4.08091

Patch 7 named wall-central y+ : min: 0.00437786 max: 3.43715 average:
    0.805145

Patch 8 named wall-chamber y+ : min: 0.042016 max: 27.7852 average:
    8.68242
```

```
Patch 9 named wall-cone y+ : min: 0.0895118 max: 5.83434 average: 2.23927

Patch 10 named wall-deflector y+ : min: 0.0675558 max: 10.6197 average:
    5.78297

Patch 11 named wall-dip y+ : min: 0.8073 max: 29.2173 average: 6.47246

Patch 12 named wall-inlet-frustum y+ : min: 0.472589 max: 10.2754 average:
     4.10162

Patch 13 named wall-inlet-large y+ : min: 0.183672 max: 3.9856 average:
    1.83187

Patch 14 named wall-inlet-small y+ : min: 3.62554 max: 8.06858 average:
    5.05902

Patch 15 named wall-overflow y+ : min: 0.775359 max: 28.824 average:
    8.84407

Patch 16 named wall-stand y+ : min: 0.000440967 max: 4.07019 average:
    0.149103

Patch 17 named wall-tray y+ : min: 0.301415 max: 14.3388 average: 5.09888

Patch 18 named wall-underflow y+ : min: 0.00373278 max: 0.672592 average:
    0.142631

Writing yPlus to field yPlus

End
```

A.4.3 Coarse Isotropic Mesh with T-Rex® Boundary Layer using Realisable k-epsilon Turbulence Model

```
Create time

Create mesh for time = 5032

Time = 5032
Calculating wall distance

Writing wall distance to field y

Reading field U

Reading/calculating face flux field phi

Selecting incompressible transport model Newtonian
Selecting RAS turbulence model realizableKE
realizableKECoeffs
{
    Cmu           0.09;
    A0            4;
    C2            1.9;
    sigmak        1;
    sigmaEps      1.2;
}

Patch 3 named stand y+ : min: 1.89412e-05 max: 5.0395e-05 average: 3.62154
    e-05

Patch 4 named underflow y+ : min: 0.0418554 max: 0.152795 average:
    0.0960631

Patch 5 named valve y+ : min: 2.17681 max: 2.38674 average: 2.24051

Patch 6 named wall-baffle y+ : min: 1.06749 max: 31.0172 average: 17.9523

Patch 7 named wall-central y+ : min: 0.217495 max: 10.6278 average:
    3.56638

Patch 8 named wall-chamber y+ : min: 0.131809 max: 154.841 average:
    51.2076
```

```
Patch 9 named wall-cone y+ : min: 1.22755 max: 23.5437 average: 10.0761

Patch 10 named wall-deflector y+ : min: 0.133439 max: 54.0272 average:
    26.7266

Patch 11 named wall-dip y+ : min: 5.36175 max: 88.5588 average: 30.1042

Patch 12 named wall-inlet-frustum y+ : min: 7.72508 max: 53.4814 average:
    19.4934

Patch 13 named wall-inlet-large y+ : min: 2.22869 max: 11.2856 average:
    9.32245

Patch 14 named wall-inlet-small y+ : min: 19.608 max: 47.8567 average:
    34.7306

Patch 15 named wall-overflow y+ : min: 3.84044 max: 92.9869 average:
    41.7241

Patch 16 named wall-stand y+ : min: 2.48794e-05 max: 7.8427 average:
    0.78587

Patch 17 named wall-tray y+ : min: 1.33319 max: 75.9193 average: 21.5523

Patch 18 named wall-underflow y+ : min: 0.0786091 max: 5.17409 average:
    1.34553

Writing yPlus to field yPlus

End
```

A.4.4 Fine Isotropic Mesh with T-Rex® Boundary Layer using Realisable k-epsilon Turbulence Model

```
Create time

Create mesh for time = 6765

Time = 6765
Calculating wall distance

Writing wall distance to field y

Reading field U

Reading/calculating face flux field phi

Selecting incompressible transport model Newtonian
Selecting RAS turbulence model realizableKE
realizableKECoeffs
{
    Cmu           0.09;
    A0            4;
    C2            1.9;
    sigmak        1;
    sigmaEps      1.2;
}

Patch 3 named stand y+ : min: 1.12462e-05 max: 5.88047e-05 average: 3.4271
    e-05

Patch 4 named underflow y+ : min: 0.0675279 max: 0.182884 average:
    0.124799

Patch 5 named valve y+ : min: 0.820856 max: 1.05896 average: 0.900028

Patch 6 named wall-baffle y+ : min: 0.624864 max: 25.3123 average: 13.5198

Patch 7 named wall-central y+ : min: 0.14355 max: 9.88528 average: 2.68768

Patch 8 named wall-chamber y+ : min: 0.0567276 max: 114.874 average:
    37.1064

Patch 9 named wall-cone y+ : min: 0.642704 max: 19.3586 average: 7.48268
```

```
Patch 10 named wall-deflector y+ : min: 0.0570037 max: 29.14 average:
    15.1202

Patch 11 named wall-dip y+ : min: 3.72611 max: 116.923 average: 23.4047

Patch 12 named wall-inlet-frustum y+ : min: 8.3917 max: 54.4289 average:
    19.7707

Patch 13 named wall-inlet-large y+ : min: 0.901627 max: 11.3875 average:
    9.64619

Patch 14 named wall-inlet-small y+ : min: 14.2484 max: 37.9022 average:
    34.0733

Patch 15 named wall-overflow y+ : min: 1.48347 max: 100.202 average:
    28.9256

Patch 16 named wall-stand y+ : min: 2.5016e-05 max: 8.65463 average:
    0.895359

Patch 17 named wall-tray y+ : min: 1.61985 max: 40.6747 average: 15.5685

Patch 18 named wall-underflow y+ : min: 0.0468889 max: 3.35006 average:
    1.17684

Writing yPlus to field yPlus

End
```

A.4.5 Coarse Isotropic Mesh with T-Rex® Boundary Layer using RNG k-epsilon Turbulence Model

```
Create time

Create mesh for time = 18831

Time = 18831
Calculating wall distance

Writing wall distance to field y

Reading field U

Reading/calculating face flux field phi

Selecting incompressible transport model Newtonian
Selecting RAS turbulence model RNGkEpsilon
RNGkEpsilonCoeffs
{
    Cmu           0.0845;
    C1            1.42;
    C2            1.68;
    sigmak        0.71942;
    sigmaEps      0.71942;
    eta0          4.38;
    beta          0.012;
}

Patch 3 named stand y+ : min: 1.89412e-05 max: 5.0395e-05 average: 3.62154
    e-05

Patch 4 named underflow y+ : min: 0.0470221 max: 0.146338 average:
    0.0950138

Patch 5 named valve y+ : min: 2.53736 max: 2.80736 average: 2.62548

Patch 6 named wall-baffle y+ : min: 2.44116 max: 31.0944 average: 21.3726

Patch 7 named wall-central y+ : min: 0.515487 max: 18.5763 average:
    7.10544

Patch 8 named wall-chamber y+ : min: 0.131431 max: 154.33 average: 50.5453
```

```
Patch 9 named wall-cone y+ : min: 2.01728 max: 26.7589 average: 14.5637

Patch 10 named wall-deflector y+ : min: 0.133352 max: 52.6622 average:
    26.7104

Patch 11 named wall-dip y+ : min: 5.43197 max: 69.399 average: 32.1719

Patch 12 named wall-inlet-frustum y+ : min: 8.90065 max: 52.1195 average:
    19.5189

Patch 13 named wall-inlet-large y+ : min: 2.57223 max: 11.1944 average:
    9.7197

Patch 14 named wall-inlet-small y+ : min: 19.4515 max: 47.2574 average:
    34.6277

Patch 15 named wall-overflow y+ : min: 3.82915 max: 66.1999 average:
    38.7674

Patch 16 named wall-stand y+ : min: 2.49573e-05 max: 8.03832 average:
    0.758295

Patch 17 named wall-tray y+ : min: 1.45957 max: 51.3255 average: 22.3093

Patch 18 named wall-underflow y+ : min: 0.0791481 max: 6.0793 average:
    1.82312

Writing yPlus to field yPlus

End
```

A.4.6 Coarse Isotropic Mesh with T-Rex® Boundary Layer using k-omega SST Turbulence Model

```
Create time

Create mesh for time = 17576

Time = 17576
Calculating wall distance

Writing wall distance to field y

Reading field U

Reading/calculating face flux field phi

Selecting incompressible transport model Newtonian
Selecting RAS turbulence model kOmegaSST
kOmegaSSTCoeffs
{
    alphaK1       0.85034;
    alphaK2       1;
    alphaOmega1   0.5;
    alphaOmega2   0.85616;
    gamma1        0.5532;
    gamma2        0.4403;
    beta1         0.075;
    beta2         0.0828;
    betaStar      0.09;
    a1            0.31;
    b1            1;
    c1            10;
    F3            false;
}

Patch 3 named stand y+ : min: 1.89412e-05 max: 5.0395e-05 average: 3.62154
    e-05

Patch 4 named underflow y+ : min: 0.000579483 max: 0.0240082 average:
    0.00797475

Patch 5 named valve y+ : min: 0.518796 max: 0.639239 average: 0.565647
```

```
Patch 6 named wall-baffle y+ : min: 0.771668 max: 31.751 average: 21.1246

Patch 7 named wall-central y+ : min: 0.0256719 max: 15.404 average:
    4.65139

Patch 8 named wall-chamber y+ : min: 0.00112574 max: 161.785 average:
    51.5277

Patch 9 named wall-cone y+ : min: 0.409732 max: 26.2065 average: 12.8996

Patch 10 named wall-deflector y+ : min: 0.00186946 max: 52.764 average:
    25.795

Patch 11 named wall-dip y+ : min: 4.57323 max: 88.2351 average: 33.3683

Patch 12 named wall-inlet-frustum y+ : min: 6.25778 max: 52.5807 average:
    17.9819

Patch 13 named wall-inlet-large y+ : min: 0.612178 max: 9.60157 average:
    7.80634

Patch 14 named wall-inlet-small y+ : min: 19.0305 max: 46.1472 average:
    33.776

Patch 15 named wall-overflow y+ : min: 2.09554 max: 78.5119 average:
    40.4699

Patch 16 named wall-stand y+ : min: 2.48794e-05 max: 7.09166 average:
    0.384716

Patch 17 named wall-tray y+ : min: 0.364304 max: 66.9396 average: 22.5394

Patch 18 named wall-underflow y+ : min: 0.00180286 max: 4.22978 average:
    0.660828

Writing yPlus to field yPlus

End
```

A.4.7 Coarse Isotropic Mesh with T-Rex® Boundary Layer using LRR Turbulence Model

```
Create time

Create mesh for time = 4521

Time = 4521
Calculating wall distance

Writing wall distance to field y

Reading field U

Reading/calculating face flux field phi

Selecting incompressible transport model Newtonian
Selecting RAS turbulence model LRR
LRRCoeffs
{
    Cmu           0.09;
    Clrr1         1.8;
    Clrr2         0.6;
    C1            1.44;
    C2            1.92;
    Cs            0.25;
    Ceps          0.15;
    sigmaEps      1.3;
    couplingFactor 0;
}

Patch 4 named underflow y+ : min: 0.012472 max: 0.0366129 average:
    0.0248852

Patch 5 named valve y+ : min: 2.80527 max: 3.16927 average: 2.92335

Patch 6 named wall-baffle y+ : min: 1.12375 max: 26.1399 average: 14.9497

Patch 7 named wall-central y+ : min: 0.0422217 max: 9.97364 average:
    3.08265

Patch 8 named wall-chamber y+ : min: 0.130905 max: 116.046 average:
    40.6176
```

```
Patch 9 named wall-cone y+ : min: 0.594572 max: 20.3004 average: 8.54049

Patch 10 named wall-deflector y+ : min: 0.13696 max: 55.2627 average:
    25.3129

Patch 11 named wall-dip y+ : min: 4.92693 max: 58.4525 average: 25.1816

Patch 12 named wall-inlet-frustum y+ : min: 9.29944 max: 53.3586 average:
    20.0906

Patch 13 named wall-inlet-large y+ : min: 2.85297 max: 11.4792 average:
    10.2779

Patch 14 named wall-inlet-small y+ : min: 20.7425 max: 47.0605 average:
    37.0767

Patch 15 named wall-overflow y+ : min: 4.39841 max: 68.9245 average:
    37.7317

Patch 16 named wall-stand y+ : min: 0.0556501 max: 8.33719 average:
    2.48537

Patch 17 named wall-tray y+ : min: 1.42102 max: 55.1938 average: 18.6381

Patch 18 named wall-underflow y+ : min: 0.0211024 max: 3.68185 average:
    0.70599

Writing yPlus to field yPlus

End
```

A.5 Mesh Comparison Study

A.5.1 Residual Plots

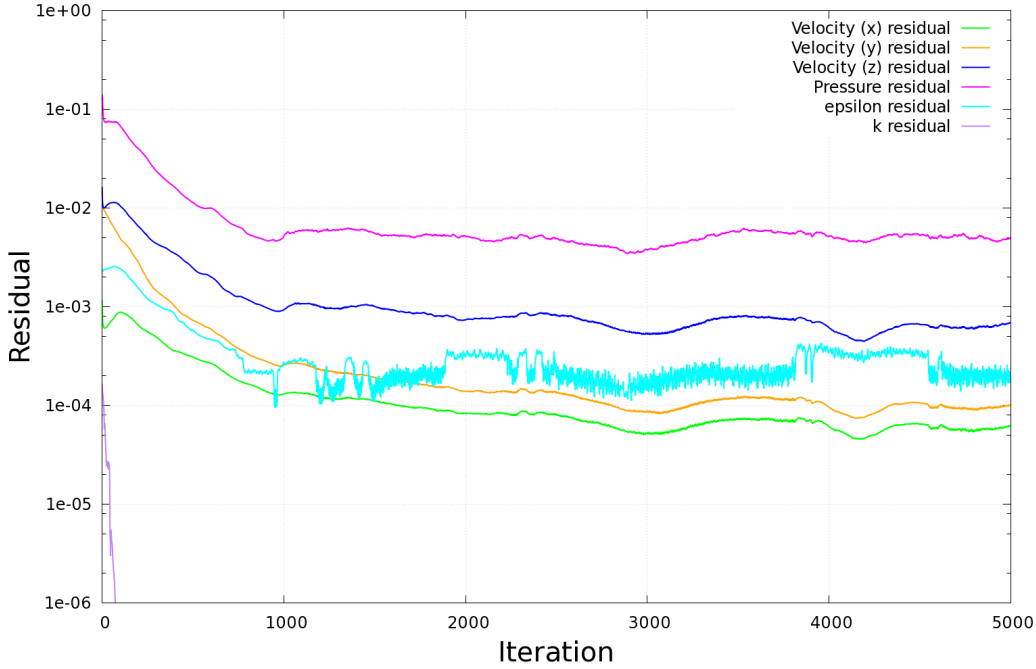


Figure A.3: Coarse isotropic mesh.

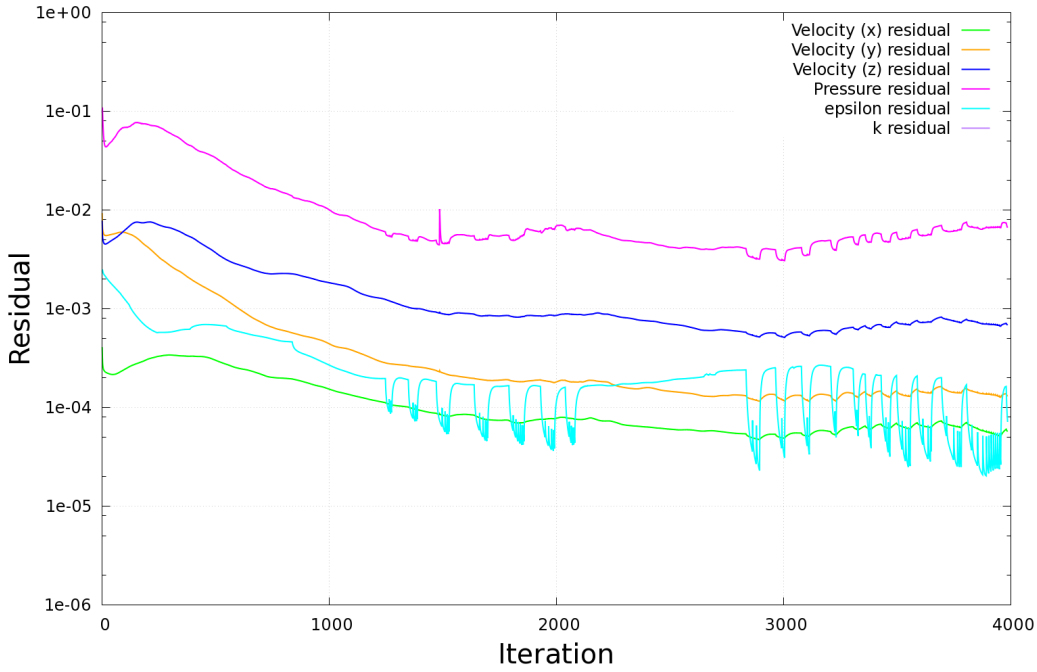


Figure A.4: Fine isotropic mesh.

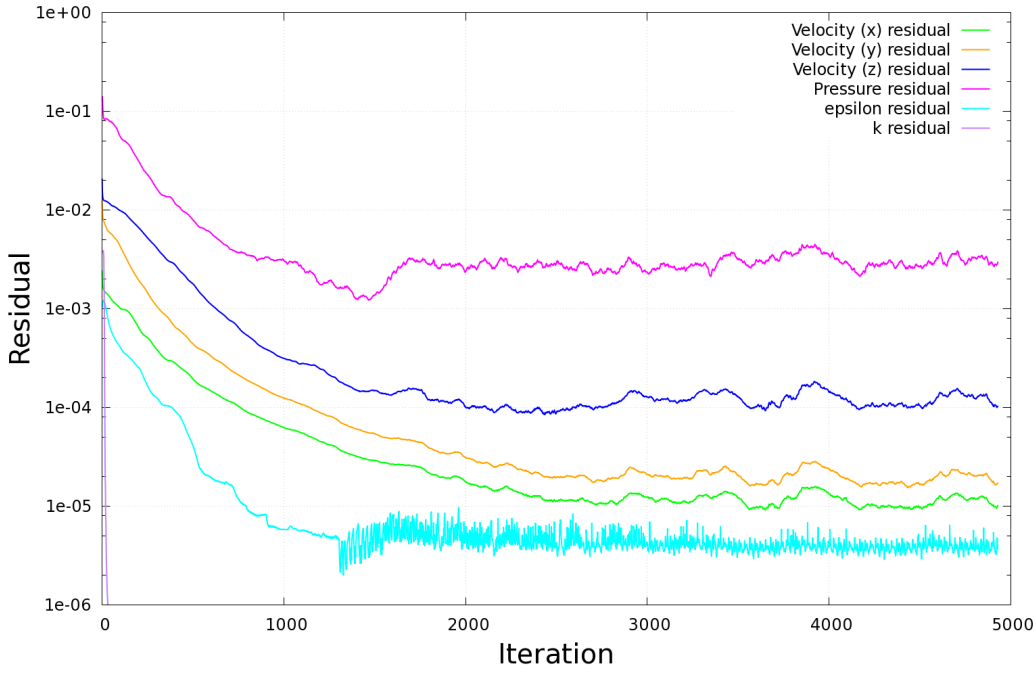


Figure A.5: Coarse isotropic mesh with anisotropic boundary layer.

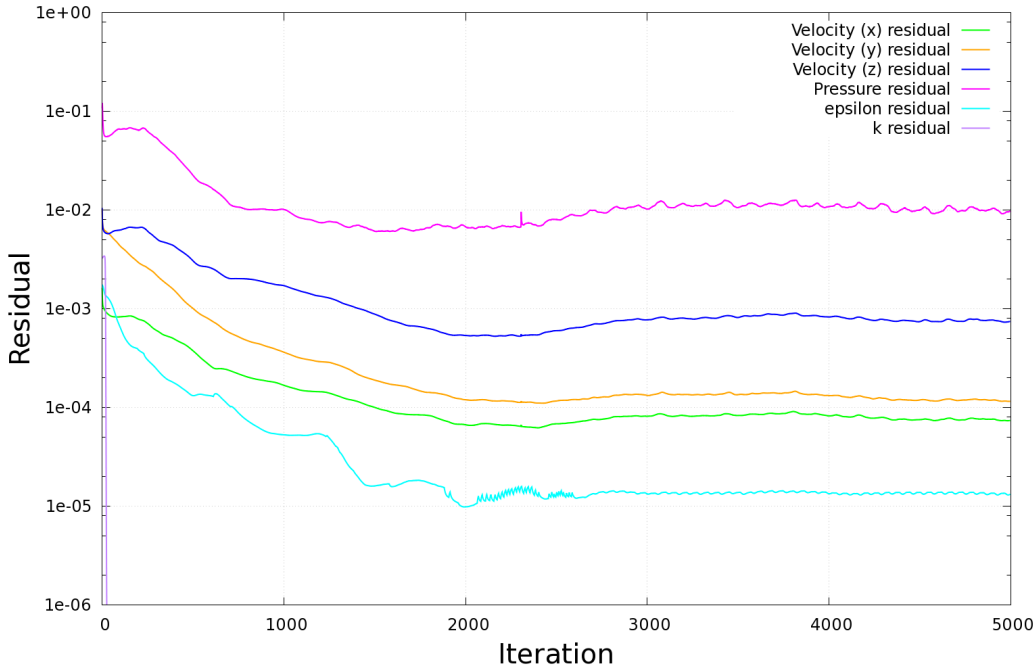


Figure A.6: Fine isotropic mesh with anisotropic boundary layer.

A.5.2 Pressure and Velocity Distributions

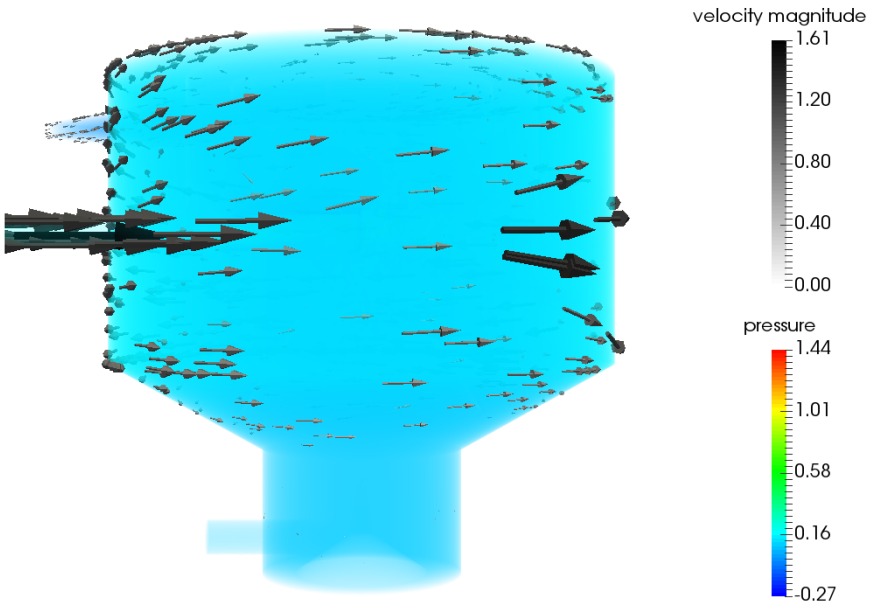


Figure A.7: Coarse isotropic mesh.

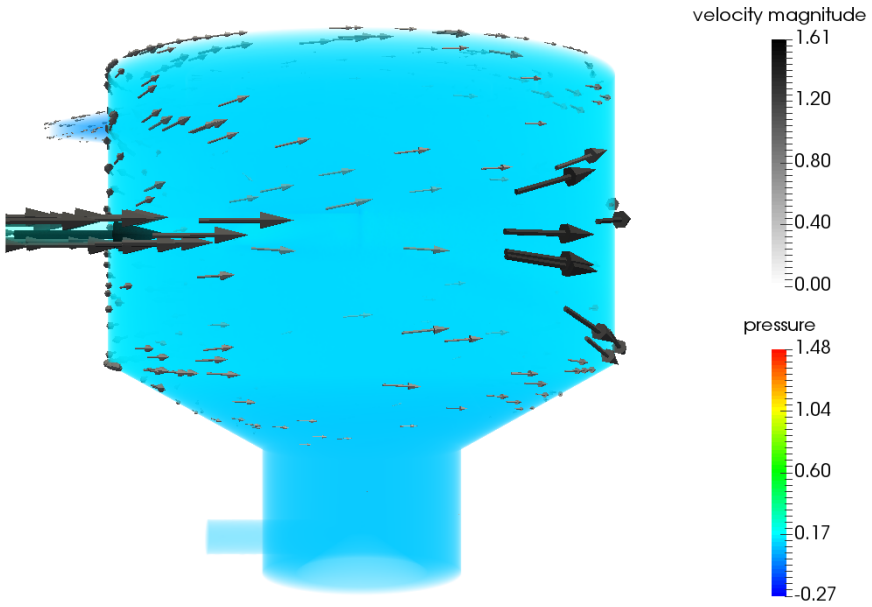


Figure A.8: Fine isotropic mesh.

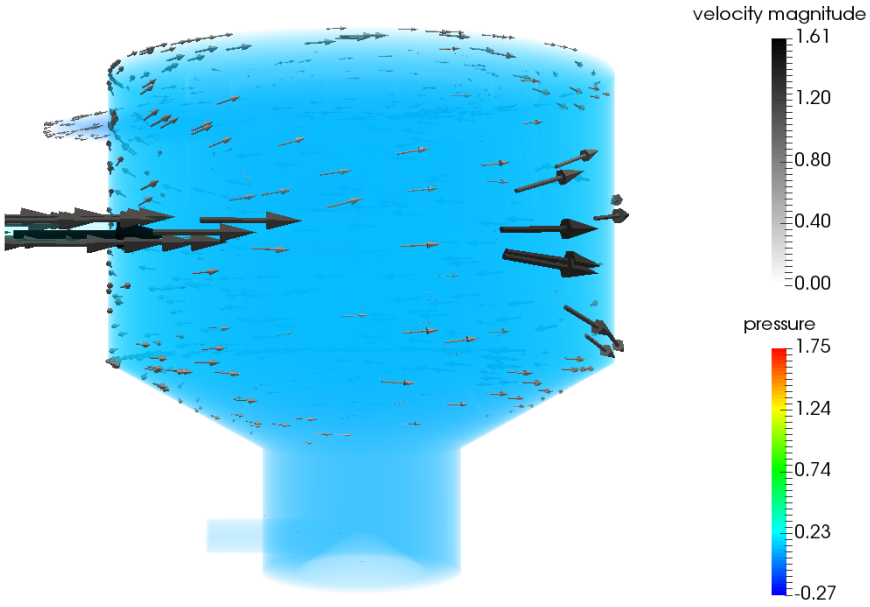


Figure A.9: Coarse isotropic mesh with anisotropic boundary layer.

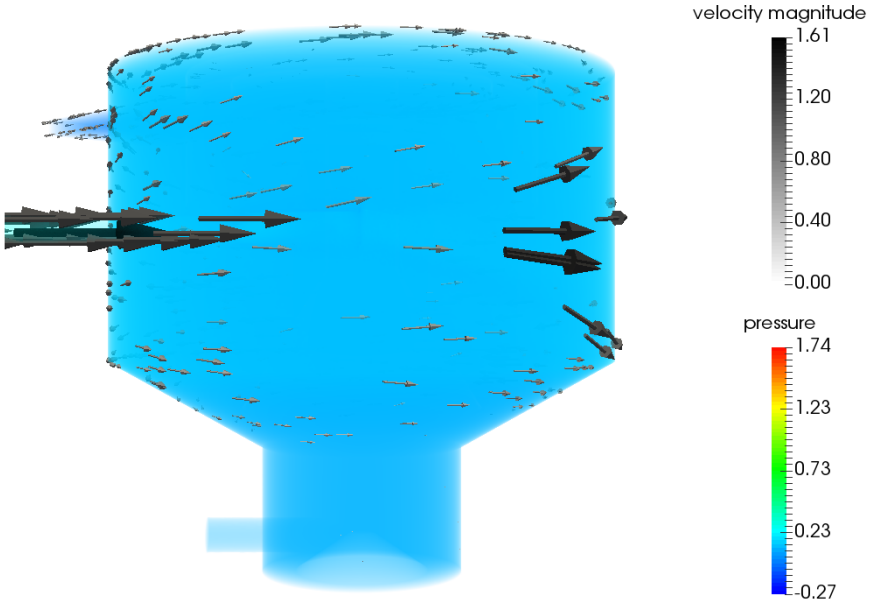


Figure A.10: Fine isotropic mesh with anisotropic boundary layer.

A.5.3 Velocity Profiles

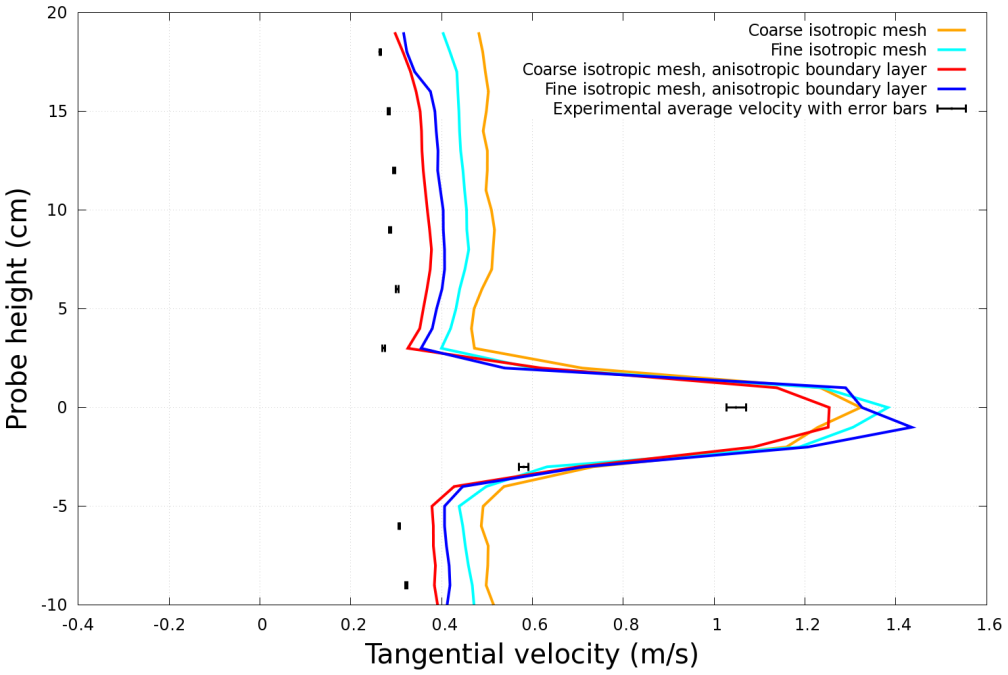


Figure A.11: First quadrant, outer ring.

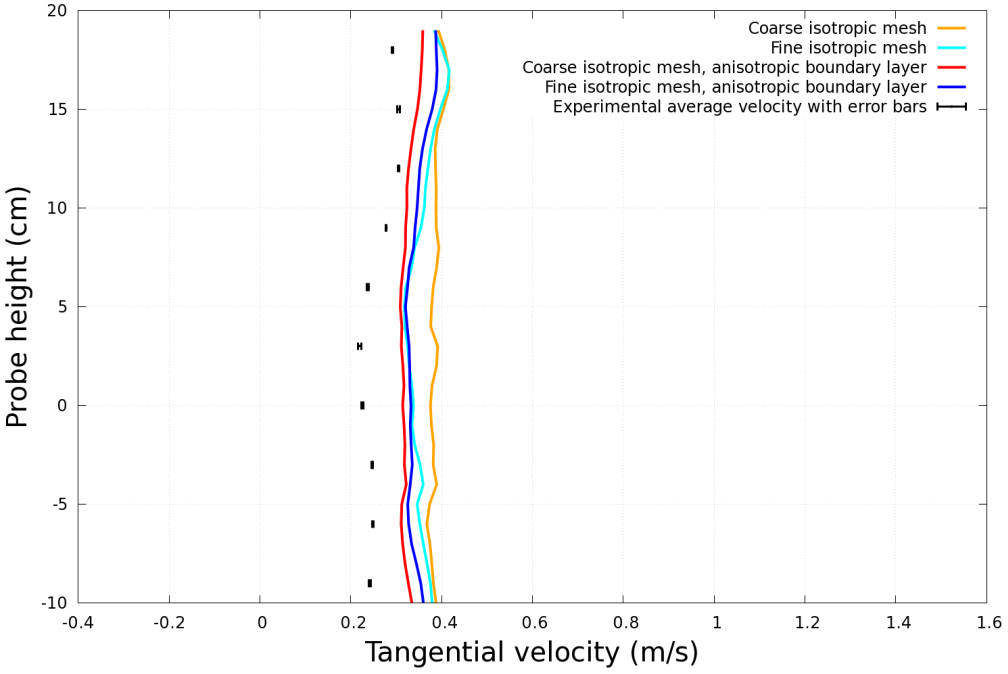


Figure A.12: First quadrant, middle ring.

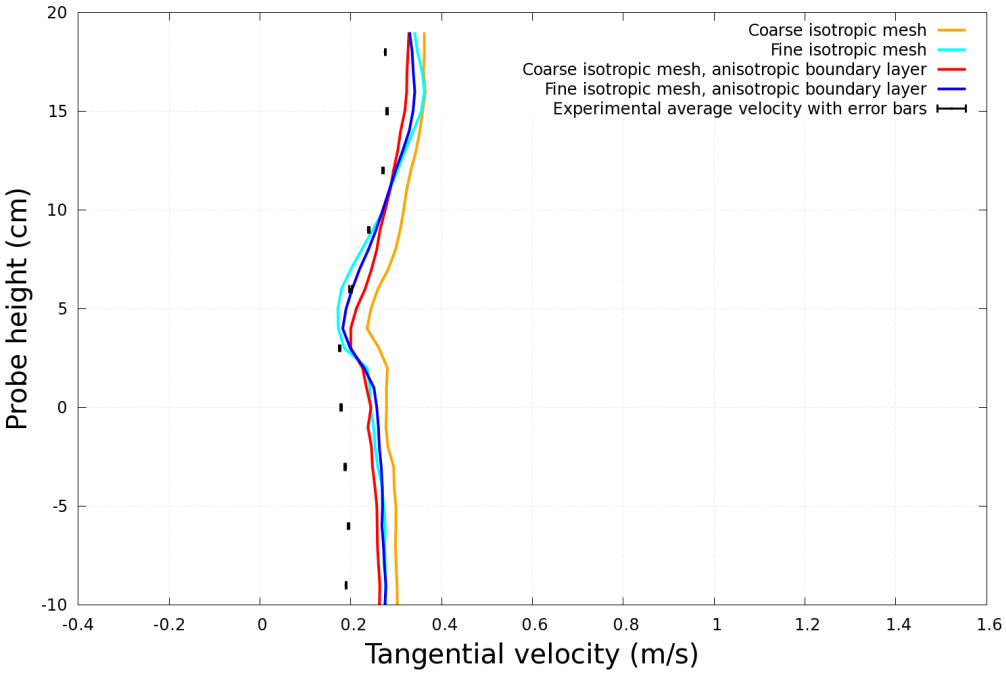


Figure A.13: First quadrant, inner ring.

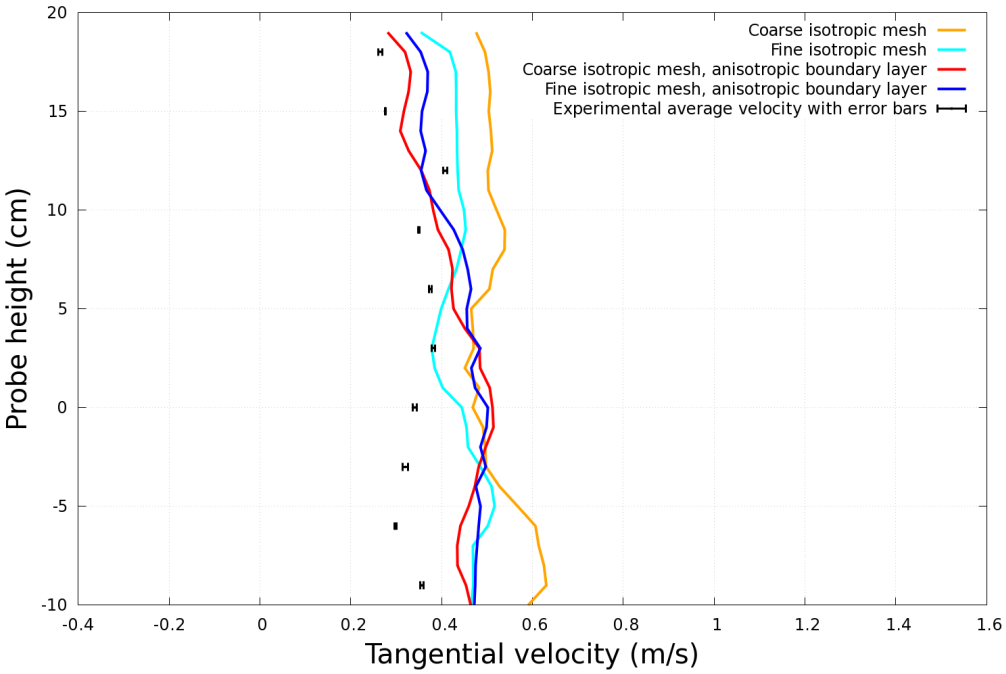


Figure A.14: Second quadrant, outer ring.

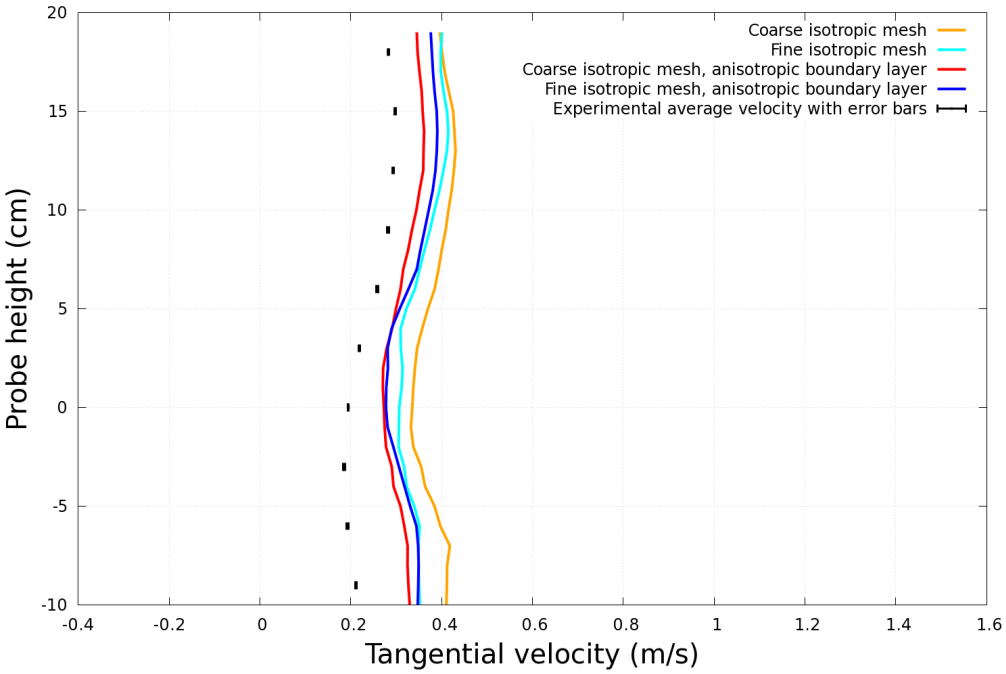


Figure A.15: Second quadrant, middle ring.

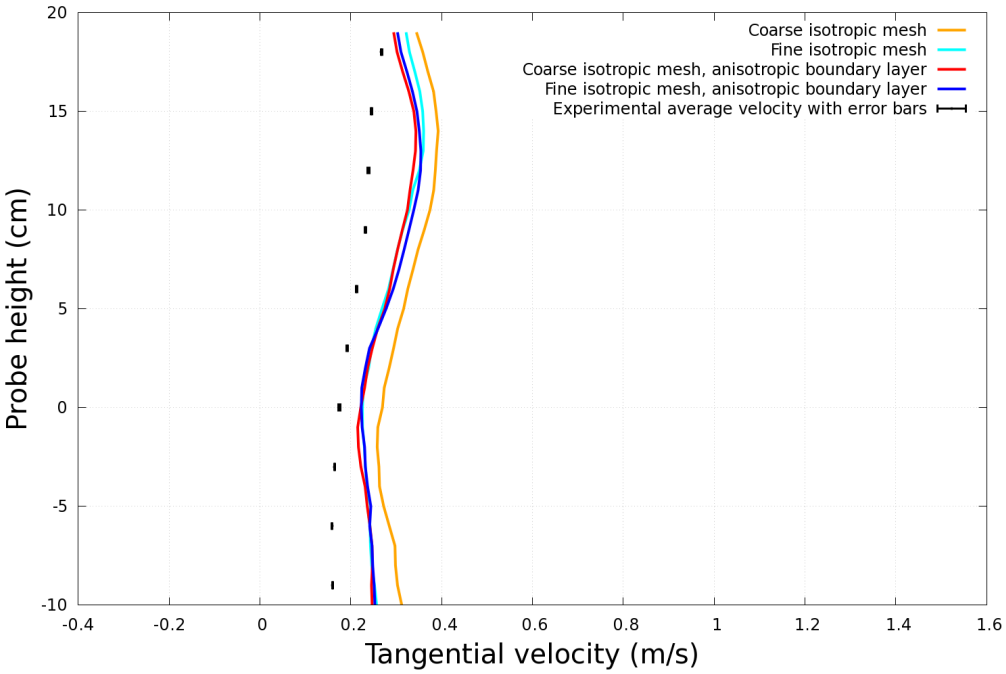


Figure A.16: Second quadrant, inner ring.

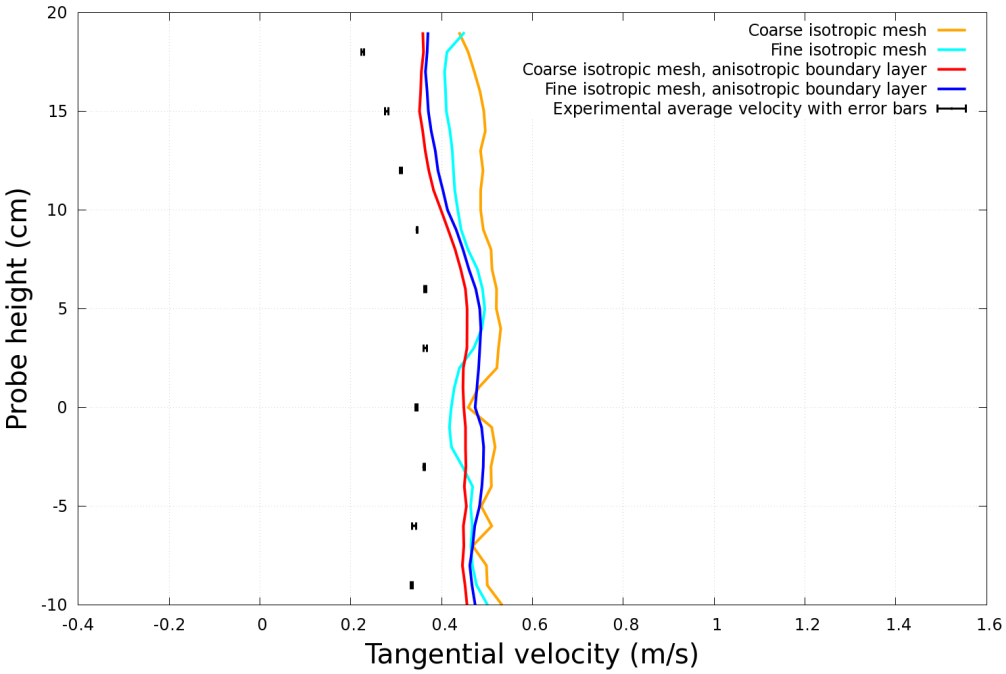


Figure A.17: Third quadrant, outer ring.

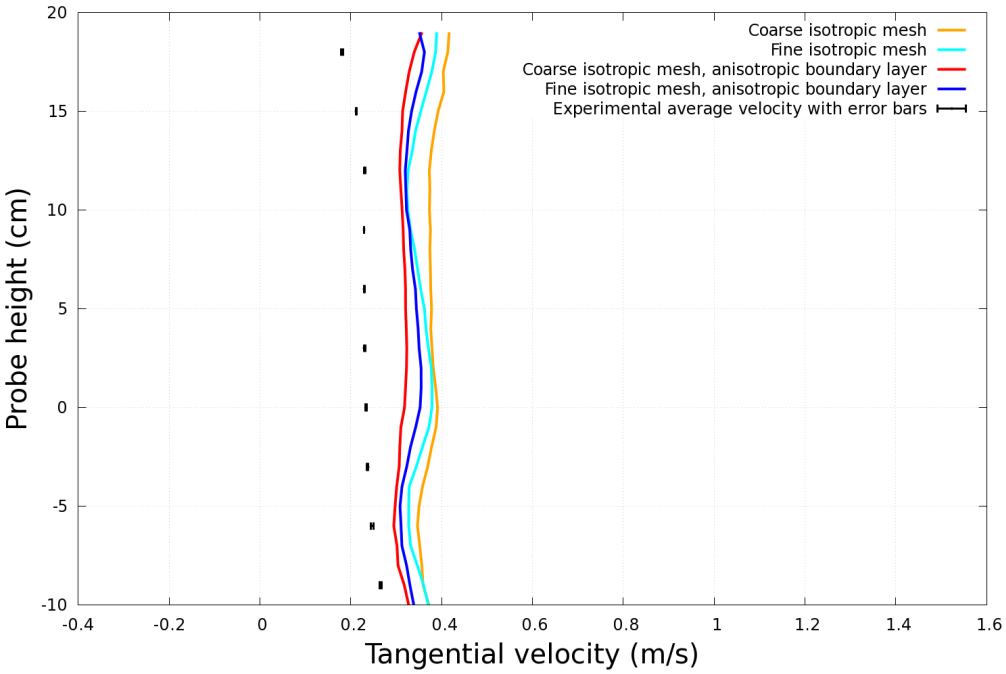


Figure A.18: Third quadrant, middle ring.

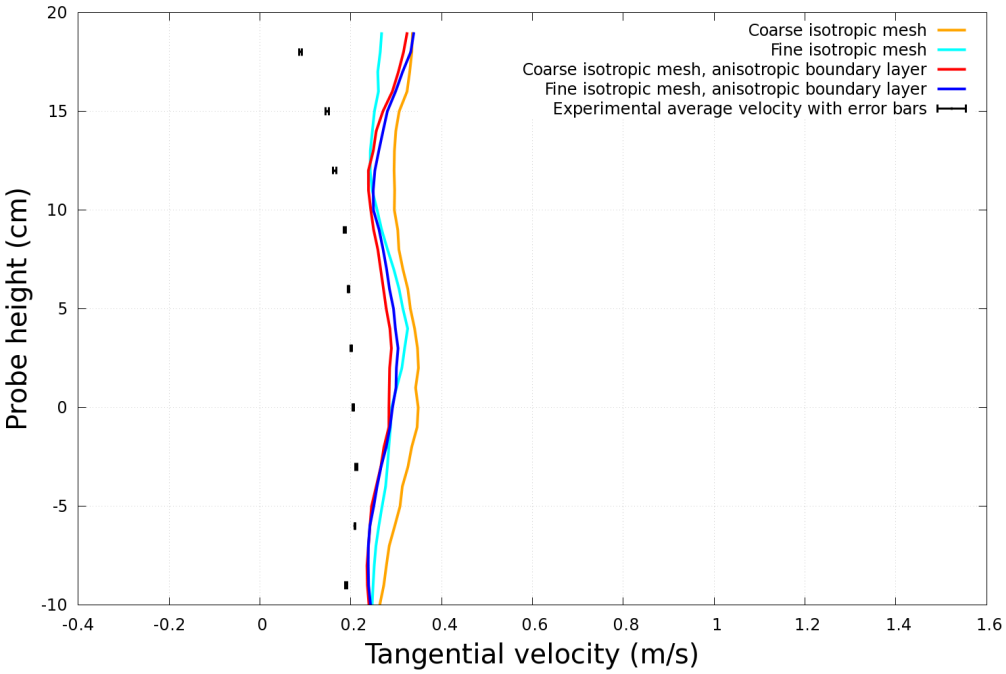


Figure A.19: Third quadrant, inner ring.

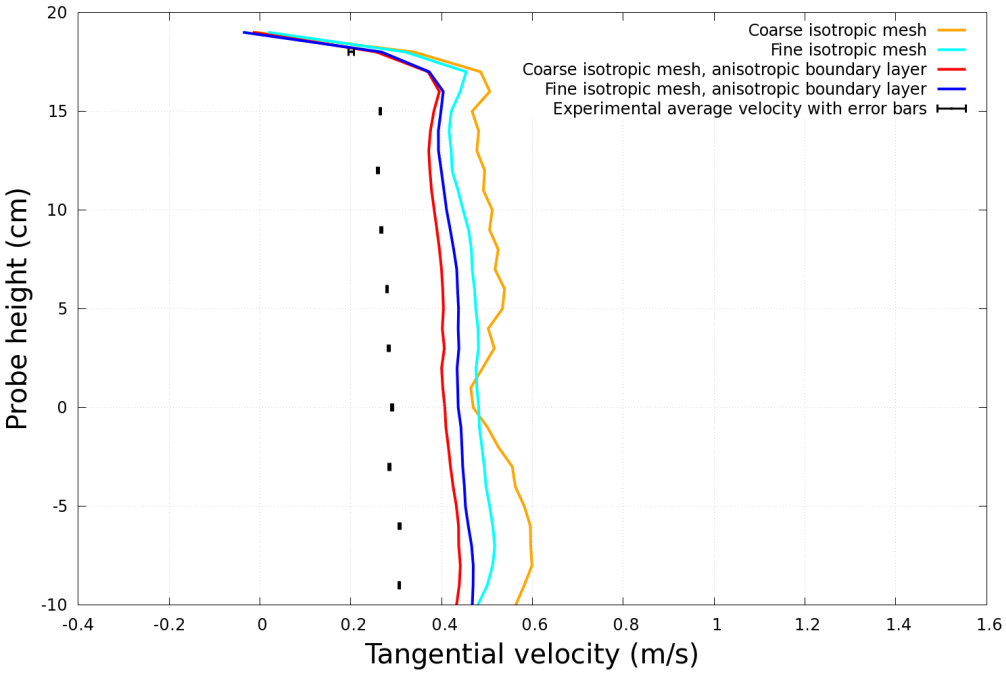


Figure A.20: Fourth quadrant, outer ring.

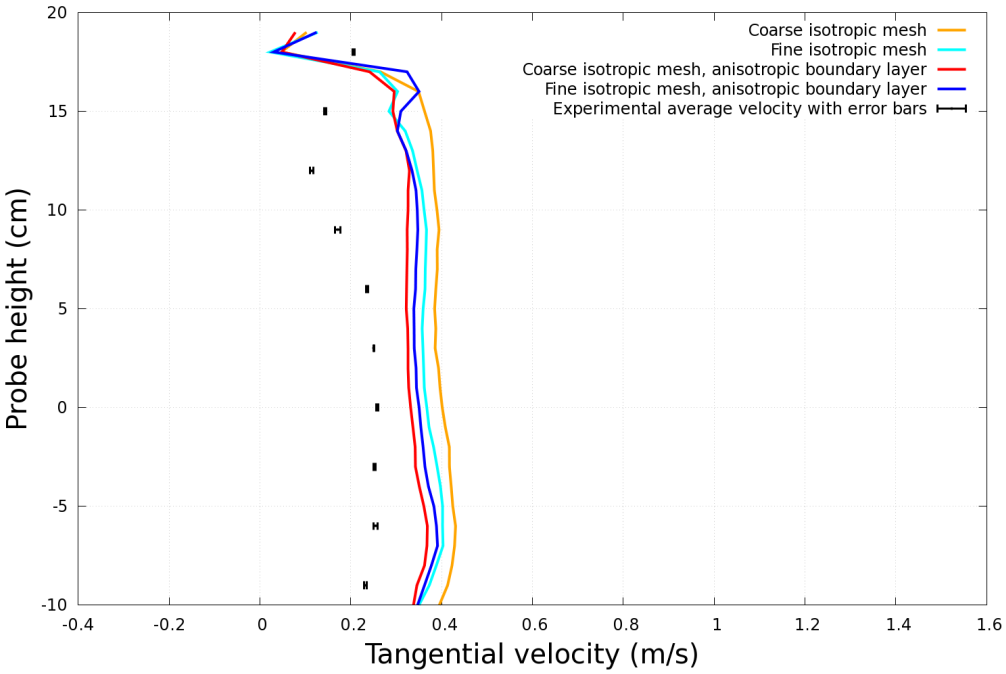


Figure A.21: Fourth quadrant, middle ring.

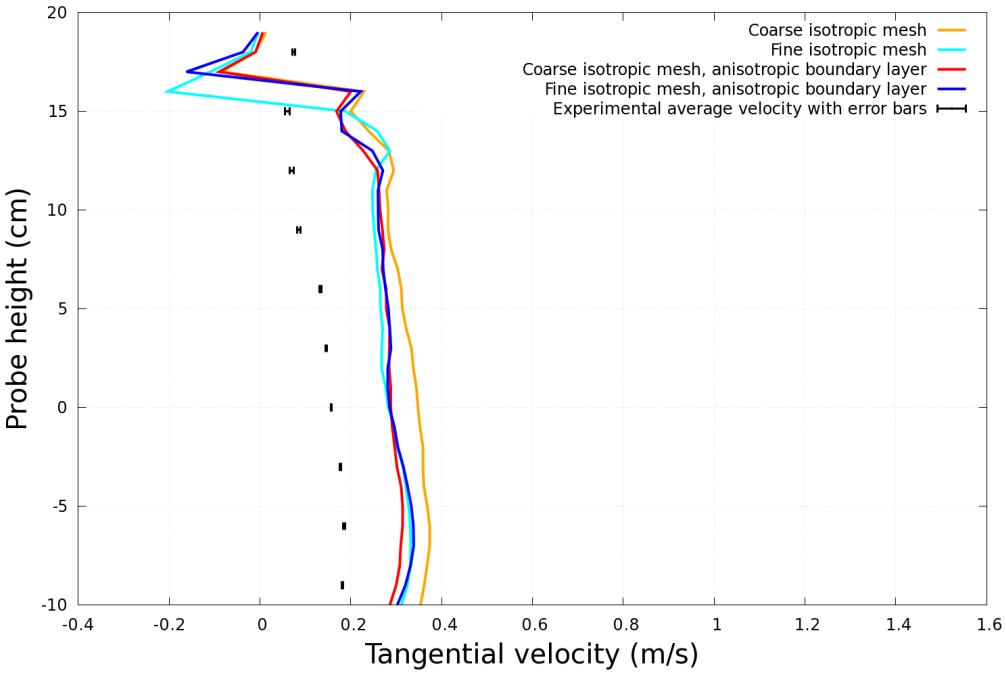


Figure A.22: Fourth quadrant, inner ring.

A.6 Turbulence Model Comparison Study

A.6.1 Residual Plots

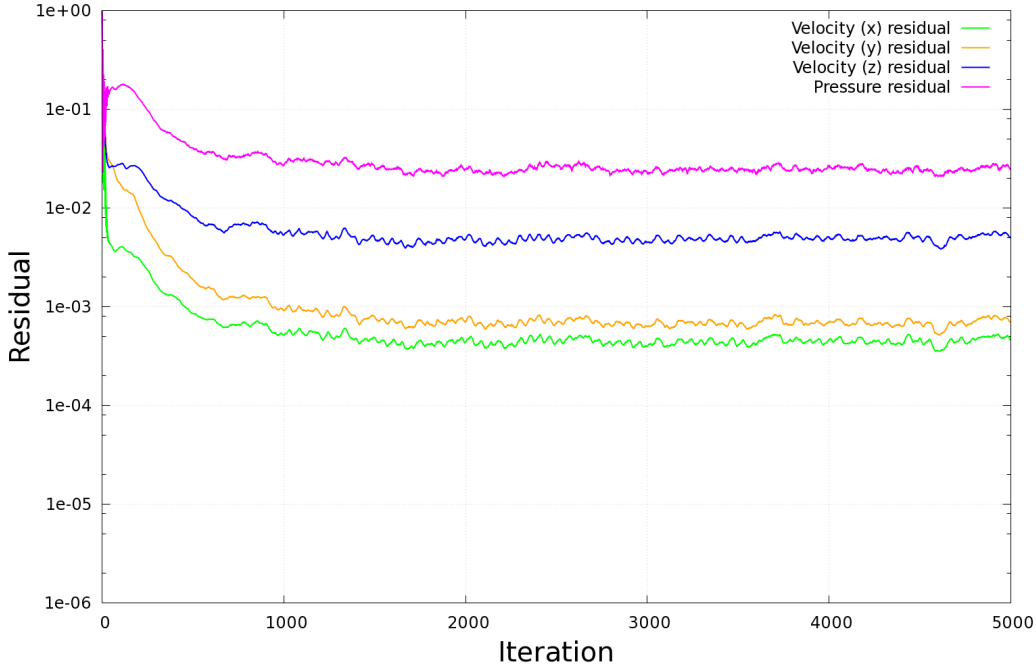


Figure A.23: No turbulence model.

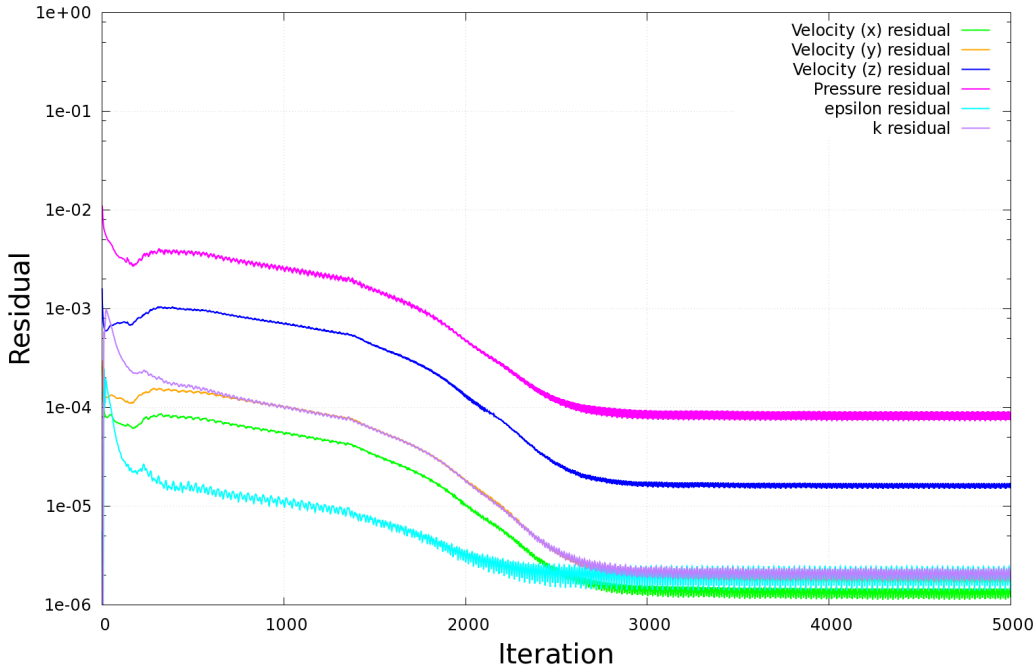


Figure A.24: RNG k-epsilon turbulence model.

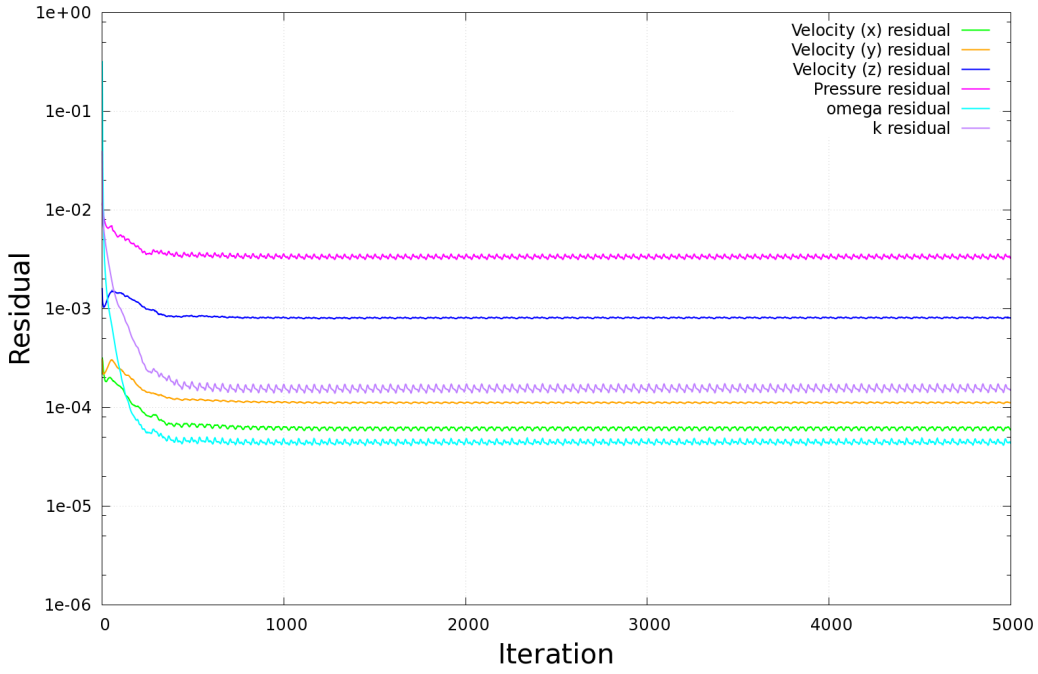


Figure A.25: k-omega turbulence model.

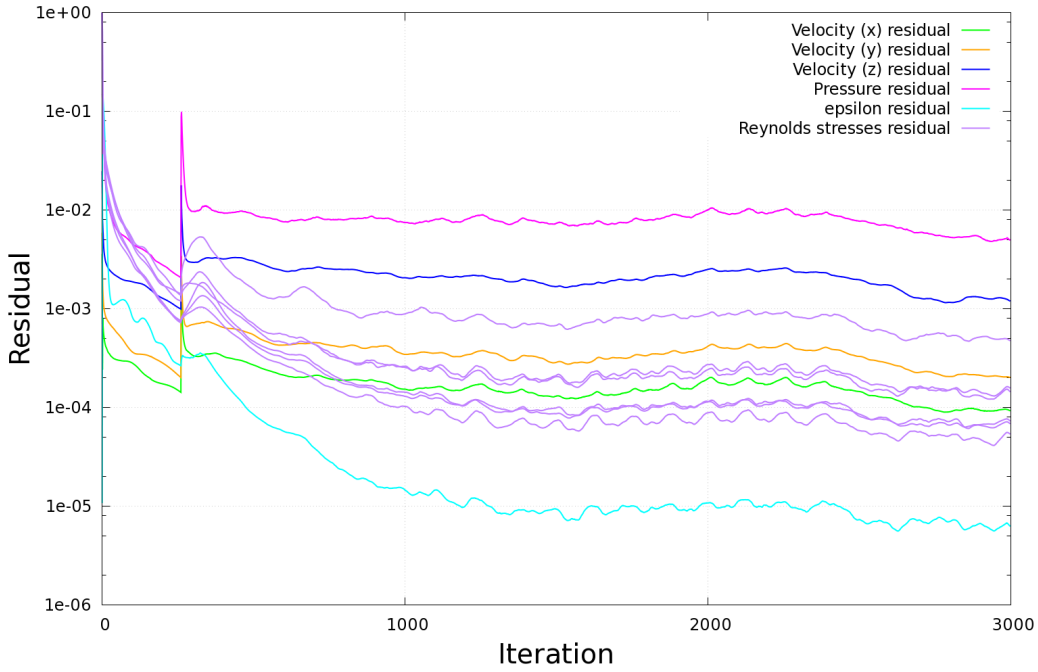


Figure A.26: LRR turbulence model.

A.6.2 Pressure and Velocity Distributions

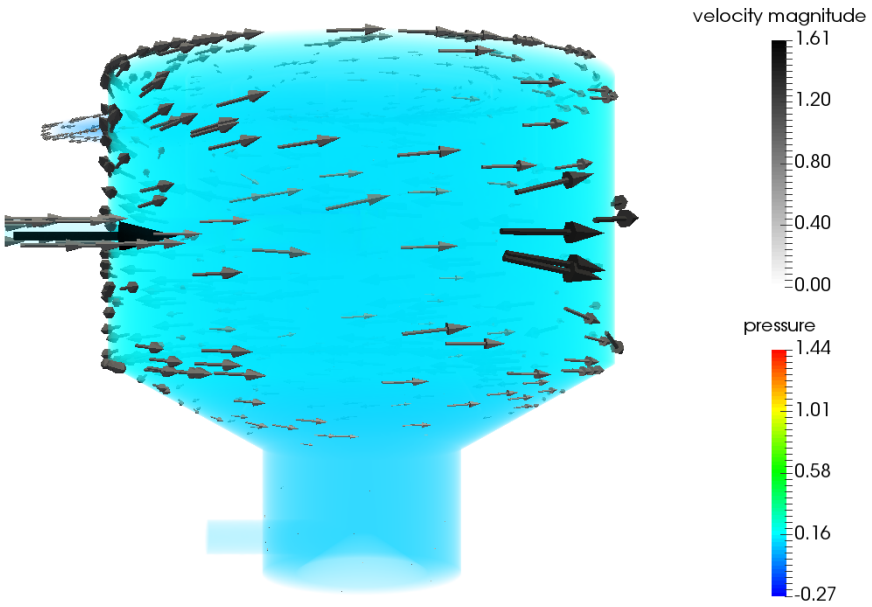


Figure A.27: No turbulence model.

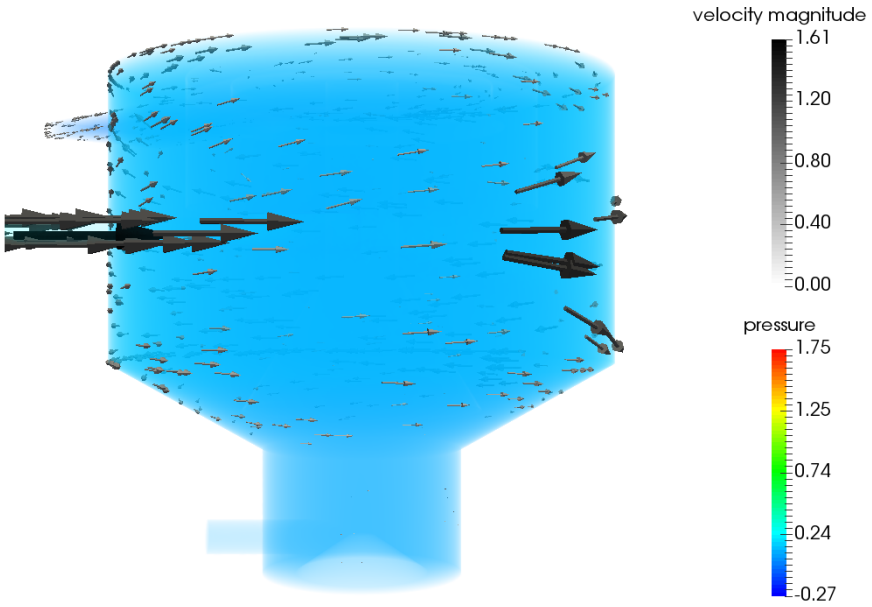


Figure A.28: RNG k-epsilon turbulence model.

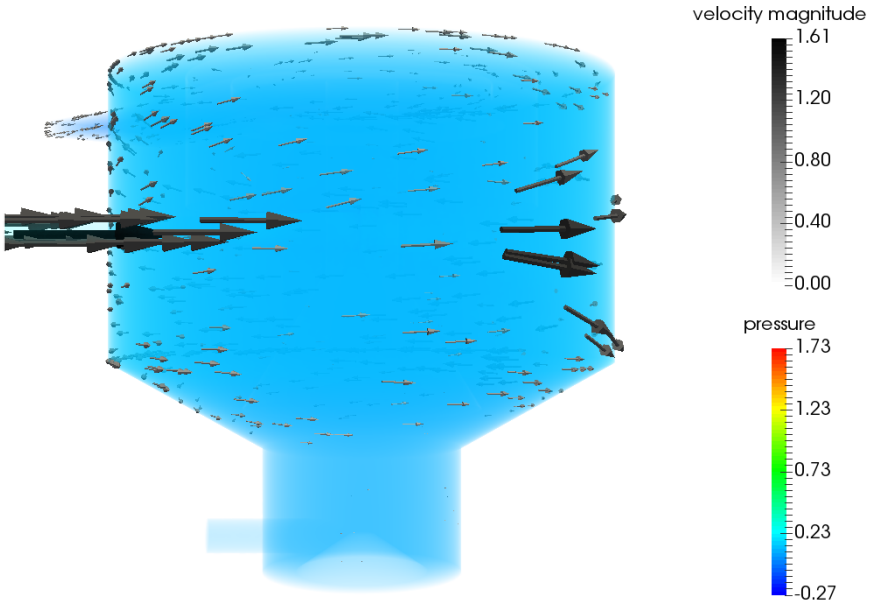


Figure A.29: k-omega turbulence model.

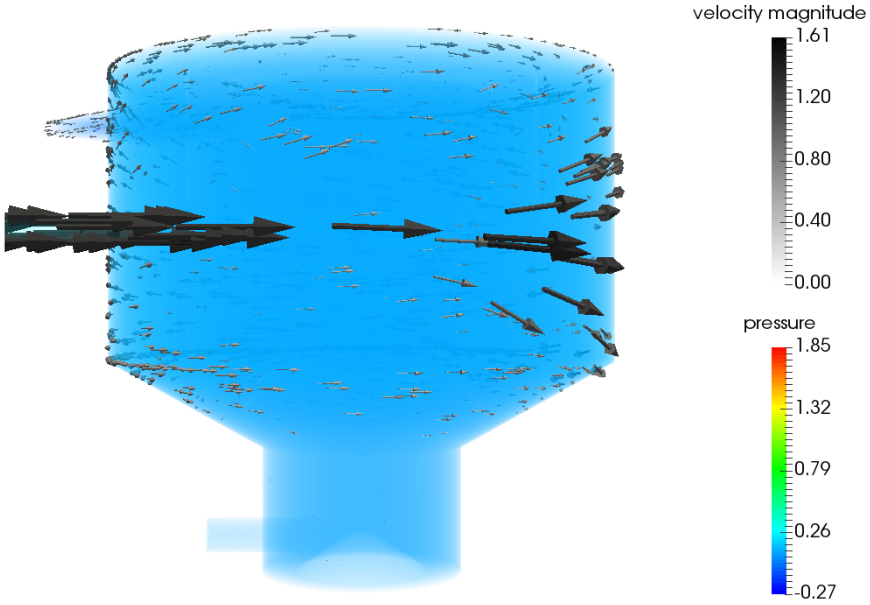


Figure A.30: LRR turbulence model.

A.6.3 Velocity Profiles

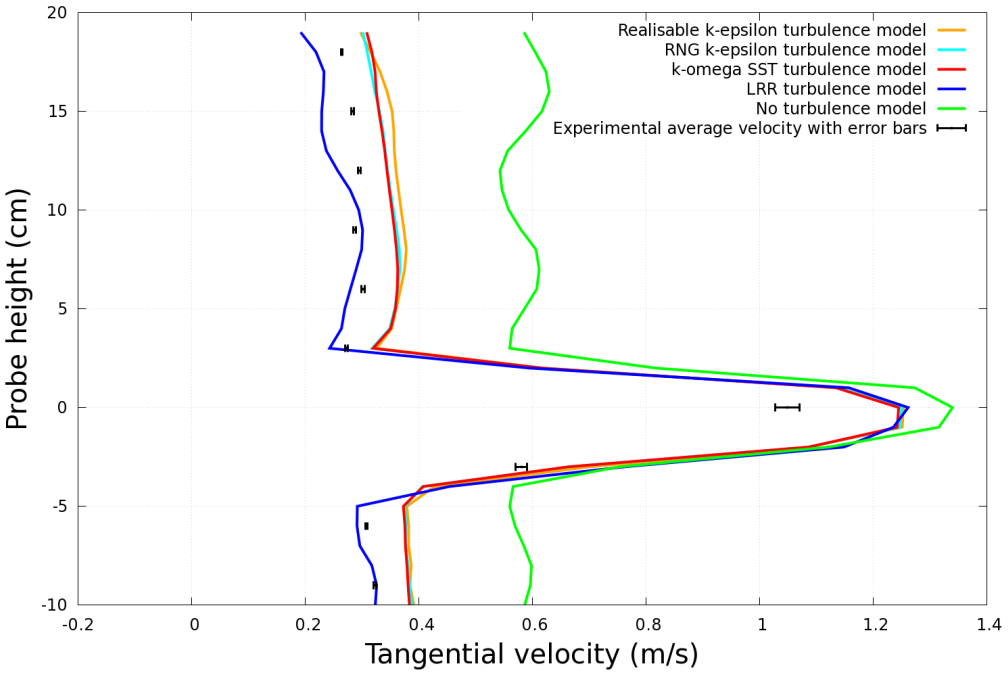


Figure A.31: First quadrant, outer ring.

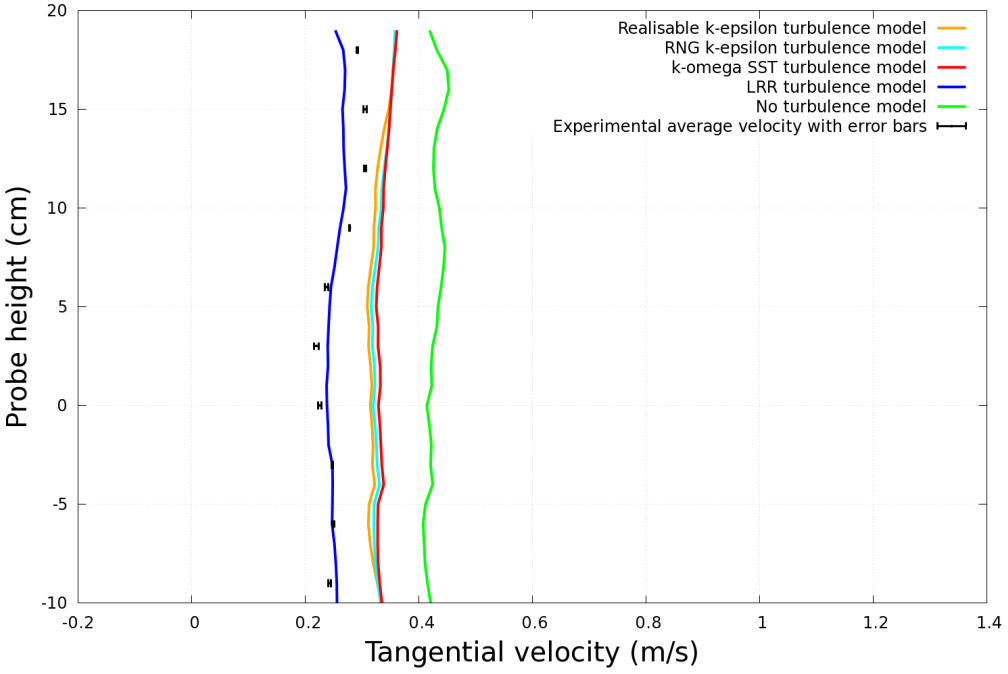


Figure A.32: First quadrant, middle ring.

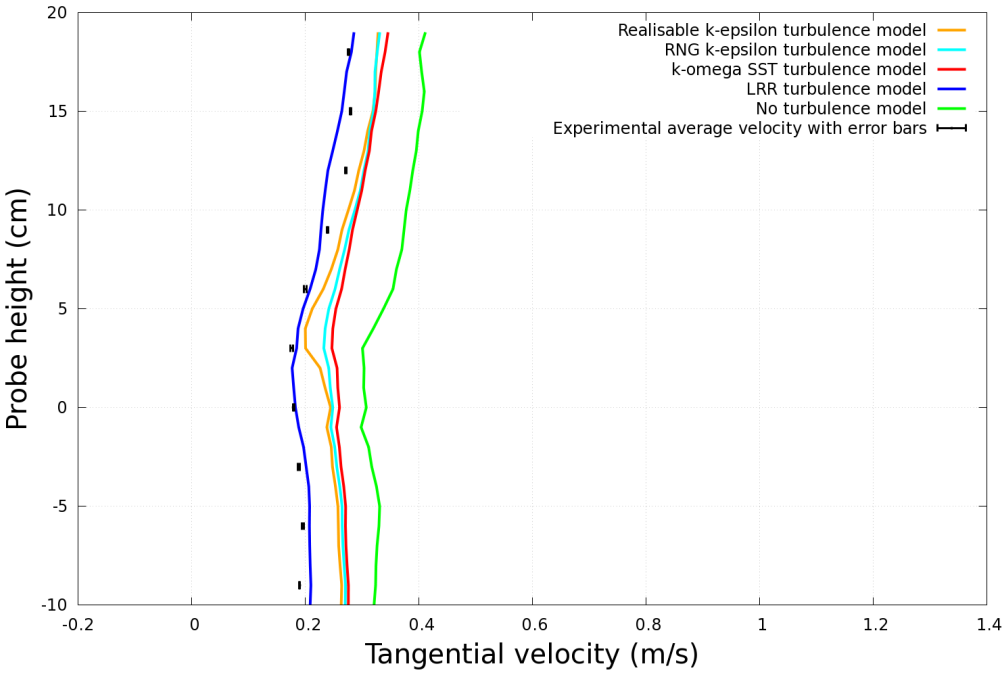


Figure A.33: First quadrant, inner ring.

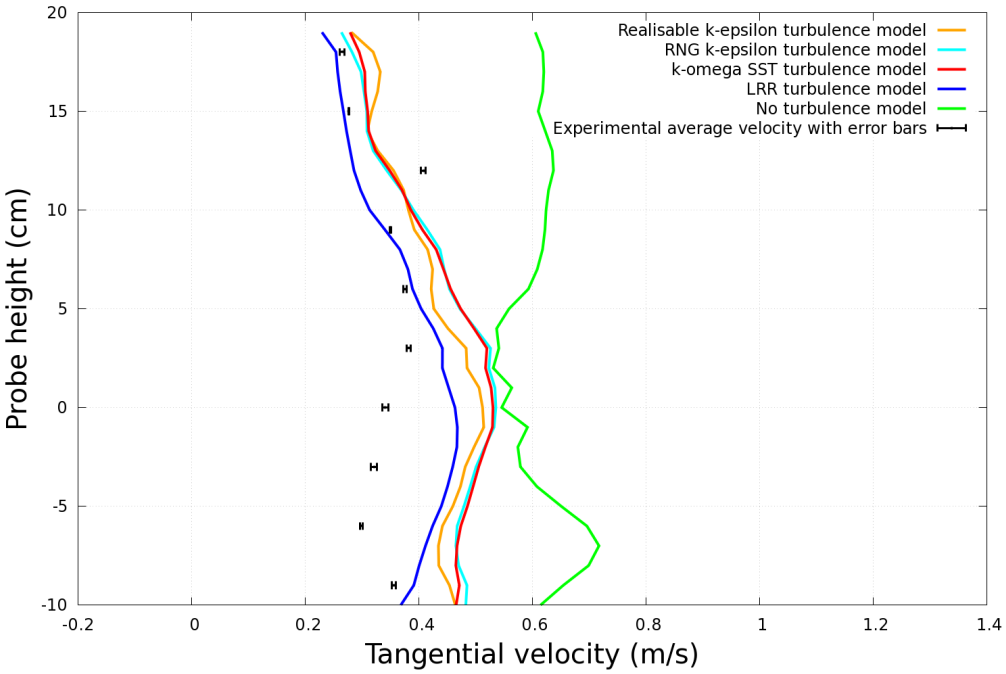


Figure A.34: Second quadrant, outer ring.

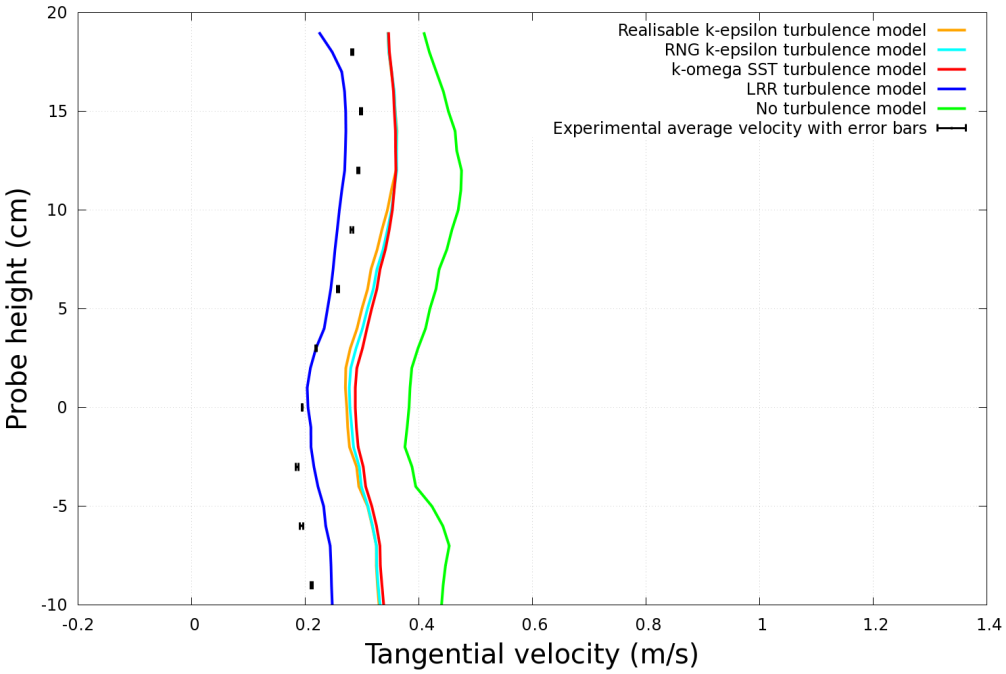


Figure A.35: Second quadrant, middle ring.

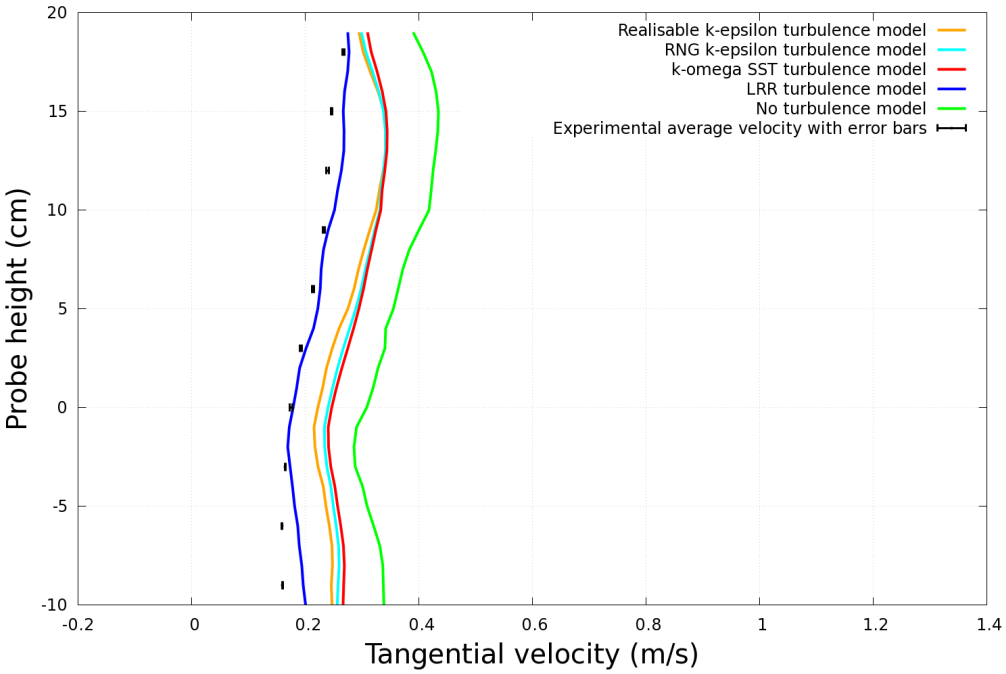


Figure A.36: Second quadrant, inner ring.

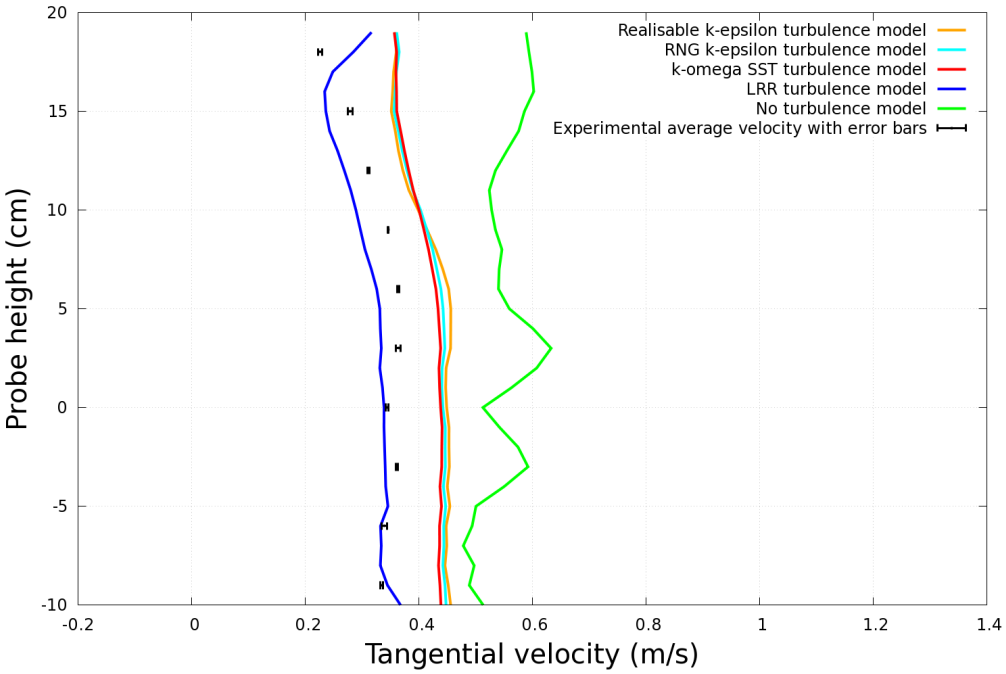


Figure A.37: Third quadrant, outer ring.

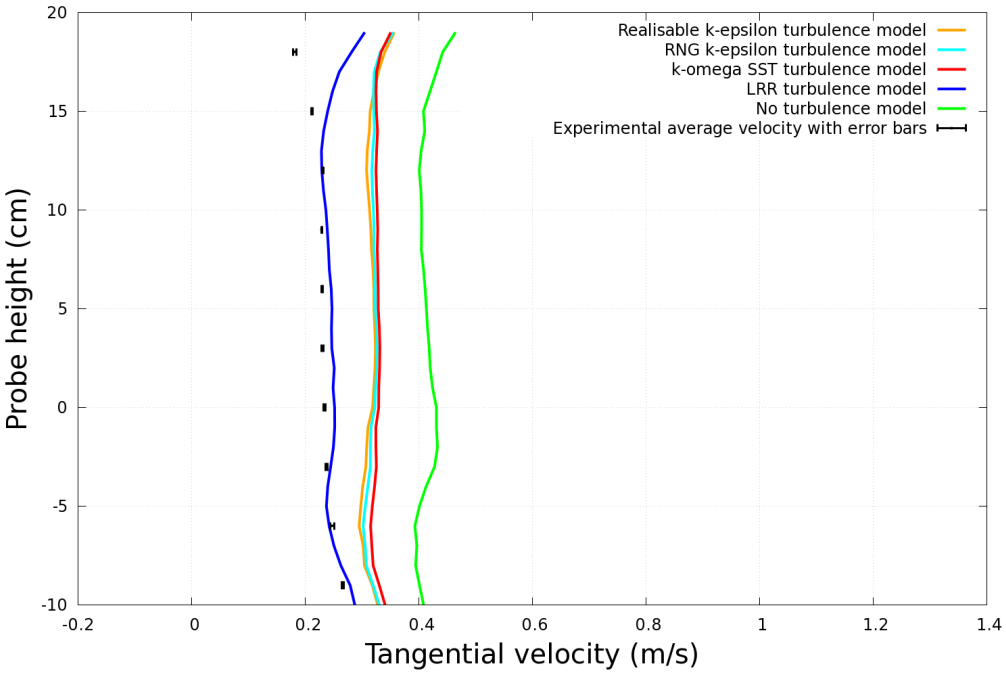


Figure A.38: Third quadrant, middle ring.

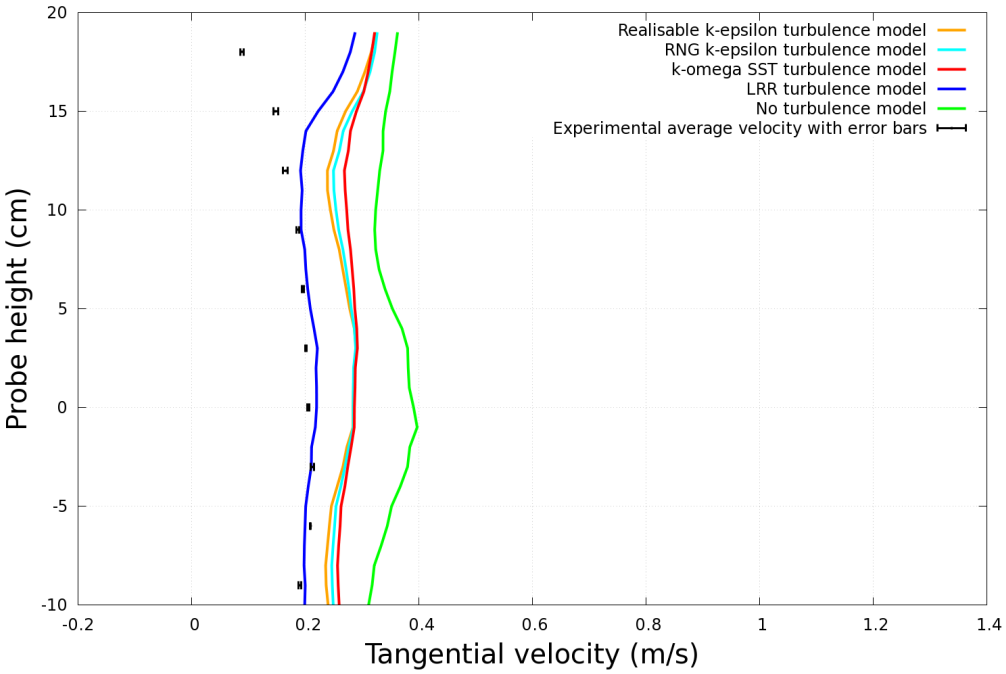


Figure A.39: Third quadrant, inner ring.

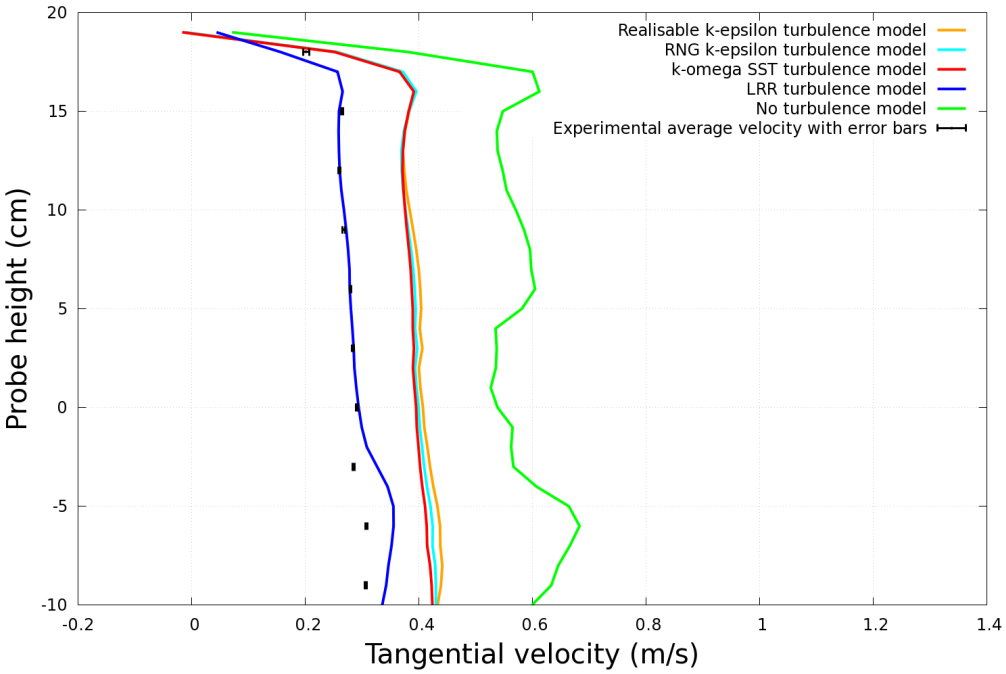


Figure A.40: Fourth quadrant, outer ring.

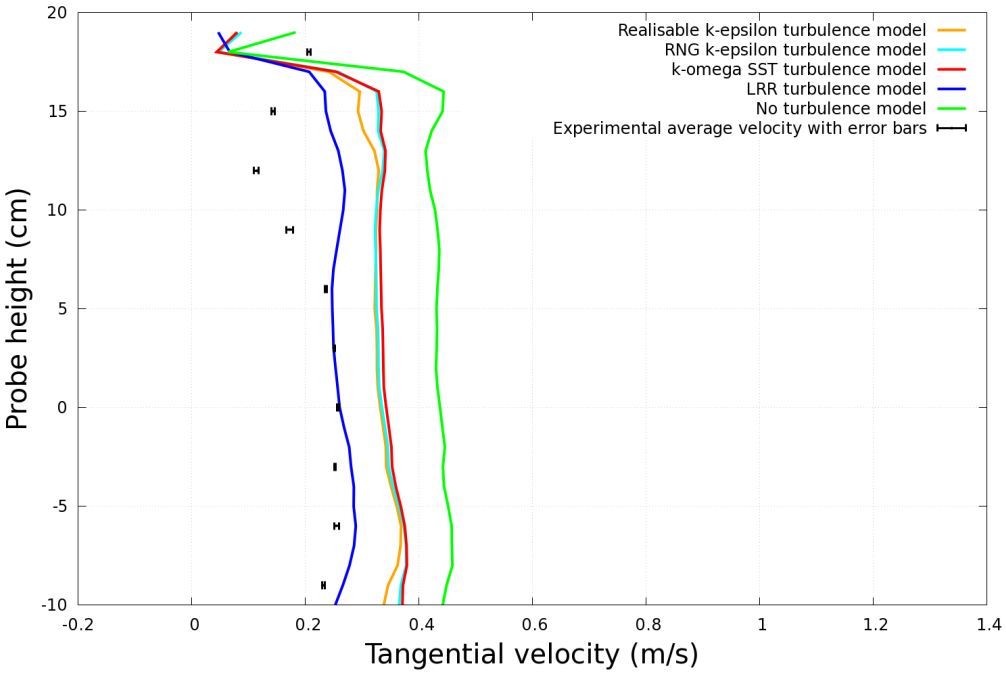


Figure A.41: Fourth quadrant, middle ring.

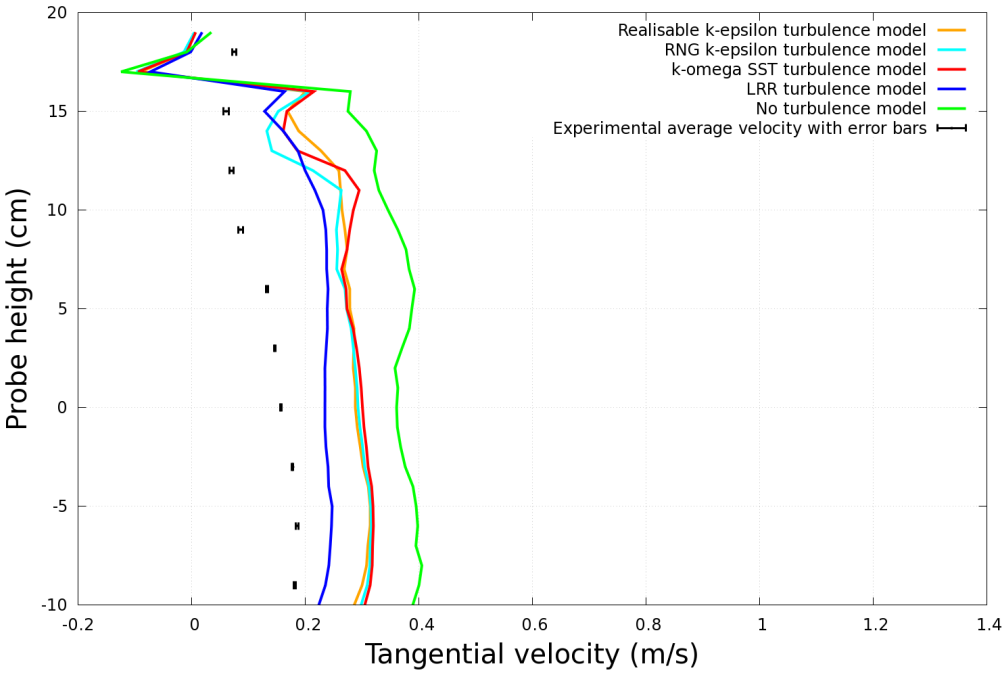


Figure A.42: Fourth quadrant, inner ring.

Appendix B

Coding

B.1 Coding the Adjoint Drift Flux Equations (4.4.2)

B.1.1 Coding Eqn. (4.4.2a)

```
volVectorField adjointTransposeConvection
(
    fvc::grad(Ua) & (U*rho)
);
zeroCells(adjointTransposeConvection, inletCells);

fvVectorMatrix UaEqn
(
    fvm::div(-phi, Ua)
  - adjointTransposeConvection
  - fvm::laplacian(muEff, Ua, "laplacian(muEff,U)")
  // - (fvc::grad(Ua) & fvc::grad(muEff))              // stability issues
  - fvc::grad(beta) * alpha
  + fvm::Sp(rho*aleph, Ua)
);

UaEqn.relax();

solve
(
    UaEqn
  ==
  - rho * fvc::reconstruct
    (
        fvc::snGrad(pa) * mesh.magSf()
    )
);
```

B.1.2 Coding Eqn. (4.4.2b)

```
{
    volScalarField rAUa(1.0/UaEqn.A());
    volVectorField HbyAa("HbyAa", Ua);
    HbyAa = rAUa*UaEqn.H();

    surfaceScalarField phiHbyAa
    (
        "phiHbyAa",
        fvc::interpolate(rho)*(fvc::interpolate(HbyAa) & mesh.Sf())
    );
    adjustPhi(phiHbyAa, Ua, pa);

    // Non-orthogonal pressure corrector loop
    while (simple.correctNonOrthogonal())
    {

        fvScalarMatrix paEqn
        (
            fvm::laplacian(sqr(rho)*rAUa, pa, "laplacian(rAUaf,pa)") == fvc
                ::div(phiHbyAa)
        );

        paEqn.setReference(paRefCell, paRefValue);
        paEqn.solve();

        if (simple.finalNonOrthogonalIter())
        {
            phia = phiHbyAa - paEqn.flux();
        }
    }

    #include "adjointContinuityErrs.H"

    // Explicitly relax pressure for adjoint momentum corrector
    pa.relax();

    // Adjoint momentum corrector
    Ua = HbyAa - rAUa*rho*fvc::grad(pa);
    Ua.correctBoundaryConditions();
}
```

B.1.3 Coding Eqn. (4.4.2c)

```
{
    surfaceScalarField phiBeta
    (
        IOobject
        (
            "phiBeta",
            runTime.timeName(),
            mesh
        ),
        phi
      - fvc::interpolate(rho)*
        (
            fvc::interpolate(Vdj) & mesh.Sf()
        )*
        (
            fvc::interpolate(alpha)*kp - 1.0
        )
    );

    surfaceScalarField phid
    (
        IOobject
        (
            "phid",
            runTime.timeName(),
            mesh
        ),
        phi*rhod/linearInterpolate(rho)
    );

    surfaceScalarField phiad
    (
        IOobject
        (
            "phiad",
            runTime.timeName(),
            mesh
        ),
        phia*rhod/linearInterpolate(rho)
    );
```

```
    fvScalarMatrix BetaEqn
    (
        fvm::div(-phiBeta, Beta)
      - fvm::laplacian(mut, Beta)
    );

    BetaEqn.relax();

    solve
    (
        BetaEqn
      ==
        (rhod - rhoc)*
        (
            fvc::div(phid,(Ua&U) - pa)
          - rhod*
            (
                Ua & (fvc::Sp(aleph,U) - g)
            )
        )
      + fvc::div(phiad)*
        (
            (rhod - rhoc)*p/rho
          - rhod*(Vdj&Vdj)*
          - (
                2.0*alpha*kp - 1.0
            )
        )
    );

    beta == Beta*rho/rhod;
}
```

B.2 Coding the Adjoint Drift Flux Boundary Conditions

B.2.1 Coding Eqn. (4.5.3)

```
void Foam::adjointOverflowPressureFvPatchScalarField::updateCoeffs()
{
    if (updated())
    {
        return;
    }

    const fvPatchField<vector>& Up =
        patch().lookupPatchField<volVectorField, vector>("U");
    const fvPatchField<vector>& Uap =
        patch().lookupPatchField<volVectorField, vector>("Ua");
    const fvPatchField<scalar>& alphap =
        patch().lookupPatchField<volScalarField, scalar>("alpha");
    const fvPatchField<scalar>& betap =
        patch().lookupPatchField<volScalarField, scalar>("beta");
    const fvPatchField<scalar>& rhop =
        patch().lookupPatchField<volScalarField, scalar>("rho");
    const fvPatchField<scalar>& muEffp =
        patch().lookupPatchField<volScalarField, scalar>("muEff");
    const dictionary& transportProperties =
        db().lookupObject<IOdictionary>("transportProperties");

    dimensionedScalar rhod(transportProperties.lookup("rhod"));
    dimensionedScalar omega(transportProperties.lookup("omega"));

    const scalarField nuEffp = muEffp/rhop;
    const scalarField Uac_ns = Uap.patchInternalField()&patch().nf();
    const scalarField delta_ns = patch().delta()&patch().nf();
    const scalarField dJUn = alphap*rhod.value();

    operator==((Up&Uap) - nuEffp*Uac_ns/delta_ns + alphap*betap/rhop
        + (omega.value()/rhop)*dJUn);

    fixedValueFvPatchScalarField::updateCoeffs();
}
```

B.2.2 Coding Eqn. (4.5.4)

```
void Foam::adjointOverflowVelocityFvPatchVectorField::updateCoeffs()
{
    if (updated())
    {
        return;
    }

    const fvPatchField<vector>& Up =
        patch().lookupPatchField<volVectorField, vector>("U");
    const fvPatchField<vector>& Uap =
        patch().lookupPatchField<volVectorField, vector>("Ua");
    const fvPatchField<scalar>& rhop =
        patch().lookupPatchField<volScalarField, scalar>("rho");
    const fvPatchField<scalar>& muEffp =
        patch().lookupPatchField<volScalarField, scalar>("muEff");

    const vectorField Uac = Uap.patchInternalField();
    const vectorField Uac_n = (Uac&patch().nf())*patch().nf();
    const vectorField Uac_t = Uac - Uac_n;

    const scalarField delta_ns = patch().delta()&patch().nf();
    const scalarField Up_ns = Up&patch().nf();
    const scalarField nuEffp = muEffp/rhop;

    vectorField Uat(Uac_t/(scalar(1) + delta_ns*Up_ns/nuEffp + VSMALL));

    vectorField::operator=(Uat);

    fixedValueFvPatchVectorField::updateCoeffs();
}
```

B.2.3 Coding Eqn. (4.5.8) with values from Eqns. (4.5.10)

```
void Foam::wallBetaFvPatchScalarField::updateCoeffs()
{
    if (this->updated())
    {
        return;
    }

    const fvPatchField<scalar>& alphap =
        patch().lookupPatchField<volScalarField, scalar>("alpha");
    const fvPatchField<scalar>& Betap =
        patch().lookupPatchField<volScalarField, scalar>("Beta");
    const fvPatchField<scalar>& rhop =
        patch().lookupPatchField<volScalarField, scalar>("rho");
    const fvPatchField<scalar>& mutp =
        patch().lookupPatchField<volScalarField, scalar>("mut");
    const fvPatchField<vector>& Vdjp =
        patch().lookupPatchField<volVectorField, vector>("Vdj");
    const dictionary& transportProperties =
        db().lookupObject<IOdictionary>("transportProperties");

    word modelType(transportProperties.lookup("VdjModel"));
    const dictionary&
        modelDict(transportProperties.subDict(modelType + "Coeffs"));
    dimensionedScalar a(modelDict.lookup("a"));
    scalar kp = a.value()*::log(10);

    const scalarField Betac = Betap.patchInternalField();
    const scalarField deltar = patch().deltaCoeffs();
    const scalarField Ep =
      - rhop*(alphap*kp - scalar(1))*(Vdjp&patch().nf());

    refValue() = scalar(0);
    refGrad() = Betac;
    valueFraction() = scalar(1)/
    (
        scalar(1)
      + deltar*mutp/(Ep + VSMALL)
    );

    mixedFvPatchScalarField::updateCoeffs();
}
```

```
void Foam::wallBetaFvPatchScalarField::operator=
(
    const fvPatchScalarField& ptf
)
{
    fvPatchScalarField::operator=
    (
       valueFraction()*refValue()
     + (1 - valueFraction())*refGrad()
    );
}
```

B.2.4 Coding Eqn. (4.5.8) with values from Eqns. (4.5.11)

```
void Foam::underflowBetaFvPatchScalarField::updateCoeffs()
{
    if (this->updated())
    {
        return;
    }

    const fvPatchField<vector>& Up =
        patch().lookupPatchField<volVectorField, vector>("U");
    const fvPatchField<vector>& Uap =
        patch().lookupPatchField<volVectorField, vector>("Ua");
    const fvPatchField<scalar>& pap =
        patch().lookupPatchField<volScalarField, scalar>("pa");
    const fvPatchField<scalar>& alphap =
        patch().lookupPatchField<volScalarField, scalar>("alpha");
    const fvPatchField<scalar>& Betap =
        patch().lookupPatchField<volScalarField, scalar>("Beta");
    const fvPatchField<scalar>& rhop =
        patch().lookupPatchField<volScalarField, scalar>("rho");
    const fvPatchField<scalar>& mutp =
        patch().lookupPatchField<volScalarField, scalar>("mut");
    const fvPatchField<vector>& Vdjp =
        patch().lookupPatchField<volVectorField, vector>("Vdj");
    const dictionary& transportProperties =
        db().lookupObject<IOdictionary>("transportProperties");

    word modelType(transportProperties.lookup("VdjModel"));
    const dictionary&
        modelDict(transportProperties.subDict(modelType + "Coeffs"));
    dimensionedScalar a(modelDict.lookup("a"));
    scalar kp = a.value()*::log(10);
    dimensionedScalar rhoc(transportProperties.lookup("rhoc"));
    dimensionedScalar rhod(transportProperties.lookup("rhod"));

    const scalarField Betac = Betap.patchInternalField();
    const scalarField Up_ns = Up&patch().nf();
    const scalarField Vdjp_ns = Vdjp&patch().nf();

    const scalarField deltar = patch().deltaCoeffs();
    const scalarField dJAlpha =
        rhod.value()*(Up_ns - (alphap*kp - scalar(1))*Vdjp_ns);
```

```
    const scalarField Dp = (rhod.value() - rhoc.value())*((Uap&Up) - pap);
    const scalarField Ep = rhop*(Up_ns - (alphap*kp - scalar(1))*Vdjp_ns);
    const scalarField Fp = -rhod.value()*(Dp*Up_ns + dJAlpha);

    refValue() = scalar(0);
    refGrad() = Betac + Fp/(deltar*mutp);
    valueFraction() = scalar(1)/
    (
        scalar(1)
      + deltar*mutp/(Ep + VSMALL)
    );

    mixedFvPatchScalarField::updateCoeffs();
}

void Foam::underflowBetaFvPatchScalarField::operator=
(
    const fvPatchScalarField& ptf
)
{
    fvPatchScalarField::operator=
    (
        valueFraction()*refValue()
      + (1 - valueFraction())*refGrad()
    );
}
```

B.2.5 Coding Eqn. (4.5.8) with values from Eqns. (4.5.12)

```
void Foam::overflowBetaFvPatchScalarField::updateCoeffs()
{
    if (this->updated())
    {
        return;
    }

    const fvPatchField<vector>& Up =
        patch().lookupPatchField<volVectorField, vector>("U");
    const fvPatchField<vector>& Uap =
        patch().lookupPatchField<volVectorField, vector>("Ua");
    const fvPatchField<scalar>& pap =
        patch().lookupPatchField<volScalarField, scalar>("pa");
    const fvPatchField<scalar>& alphap =
        patch().lookupPatchField<volScalarField, scalar>("alpha");
    const fvPatchField<scalar>& Betap =
        patch().lookupPatchField<volScalarField, scalar>("Beta");
    const fvPatchField<scalar>& rhop =
        patch().lookupPatchField<volScalarField, scalar>("rho");
    const fvPatchField<scalar>& mutp =
        patch().lookupPatchField<volScalarField, scalar>("mut");
    const fvPatchField<vector>& Vdjp =
        patch().lookupPatchField<volVectorField, vector>("Vdj");
    const dictionary& transportProperties =
        db().lookupObject<IOdictionary>("transportProperties");

    word modelType(transportProperties.lookup("VdjModel"));
    const dictionary&
        modelDict(transportProperties.subDict(modelType + "Coeffs"));
    dimensionedScalar a(modelDict.lookup("a"));
    scalar kp = a.value()*::log(10);
    dimensionedScalar rhoc(transportProperties.lookup("rhoc"));
    dimensionedScalar rhod(transportProperties.lookup("rhod"));
    dimensionedScalar omega(transportProperties.lookup("omega"));

    const scalarField Betac = Betap.patchInternalField();
    const scalarField Up_ns = Up&patch().nf();
    const scalarField Vdjp_ns = Vdjp&patch().nf();

    const scalarField deltar = patch().deltaCoeffs();
    const scalarField dJAlpha =
```

```
        rhod.value()*(Up_ns - (alphap*kp - scalar(1))*Vdjp_ns);
    const scalarField Dp = (rhod.value() - rhoc.value())*((Uap&Up) - pap);
    const scalarField Ep = rhop*(Up_ns - (alphap*kp - scalar(1))*Vdjp_ns);
    const scalarField Fp =
      - rhod.value()*(Dp*Up_ns + omega.value()*dJAlpha);

    refValue() = scalar(0);
    refGrad() = Betac + Fp/(deltar*mutp);
    valueFraction() = scalar(1)/
    (
        scalar(1)
      + deltar*mutp/(Ep + VSMALL)
    );

    mixedFvPatchScalarField::updateCoeffs();
}

void Foam::overflowBetaFvPatchScalarField::operator=
(
    const fvPatchScalarField& ptf
)
{
    fvPatchScalarField::operator=
    (
        valueFraction()*refValue()
      + (1 - valueFraction())*refGrad()
    );
}
```

Appendix C

Mesh Statistics for Structured Mesh

```
Create time

Create polyMesh for time = 0

Time = 0

Mesh stats
    points:         1831720
    faces:          5436973
    internal faces: 5379851
    cells:          1802804
    faces per cell: 6
    boundary patches: 5
    point zones:    0
    face zones:     0
    cell zones:     0

Overall number of cells of each type:
    hexahedra:   1802804
    prisms:      0
    wedges:      0
    pyramids:    0
    tet wedges:  0
    tetrahedra:  0
    polyhedra:   0

Checking topology...
    Boundary definition OK.
    Cell to face addressing OK.
    Point usage OK.
    Upper triangular ordering OK.
```

```
    Face vertices OK.
    Number of regions: 1 (OK).

Checking patch topology for multiply connected surfaces...
    Patch             Faces   Points  Surface topology
    free              13502   13605   ok (non-closed singly connected)
    inlet             64      73      ok (non-closed singly connected)
    overflow          162     190     ok (non-closed singly connected)
    underflow         189     204     ok (non-closed singly connected)
    walls             43205   43337   ok (non-closed singly connected)

Checking geometry...
    Overall domain bounding box (-1 -0.15 -0.3) (0.350085 0.44 0.3)
    Mesh (non-empty, non-wedge) directions (1 1 1)
    Mesh (non-empty) directions (1 1 1)
    Boundary openness (7.85718e-17 -1.80628e-15 3.93197e-17) OK.
    Max cell openness = 3.63998e-16 OK.
    Max aspect ratio = 23.3424 OK.
    Minimum face area = 1.83877e-06. Maximum face area = 0.000238758. Face
         area magnitudes OK.
    Min volume = 3.92743e-09. Max volume = 5.09027e-07. Total volume =
        0.13202. Cell volumes OK.
    Mesh non-orthogonality Max: 76.8786 average: 5.1918
   *Number of severely non-orthogonal (> 70 degrees) faces: 89.
    Non-orthogonality check OK.
  <<Writing 89 non-orthogonal faces to set nonOrthoFaces
    Face pyramids OK.
    Max skewness = 2.71376 OK.
    Coupled point location match (average 0) OK.

Mesh OK.

End
```

Bibliography

Andoh, R., Carter, H., and Raymond, N. The use of Advanced Vortex Technologies for CSO Control, Treatment and Disinfection at Saco, ME WWTP. In *12th International Conference on Urban Drainage*, Porto Alegre, Brazil, 2011.

Andoh, R. Y. G. and Smisson, R. P. M. High Rate Sedimentation in Hydrodynamic Separators. In *2nd International Conference on Hydraulic Modelling*, pages 341–358, 1993.

Andoh, R. Y. G., Hides, S. P., and Saul, A. J. Improving Water Quality Using Hydrodynamic Vortex Separators and Screening Systems. In *Global Solutions for Urban Drainage*, 2002.

Arceivala, S. J. and Asolekar, S. R. *Wastewater Treatment for Pollution Control and Reuse*. Tata McGraw-Hill, New Delhi, Third edition, 2008.

Bingham, E. *Fluidity and Plasticity*. New York: McGraw-Hill, First edition, 1922.

Borrvall, T. and Petersson, J. Topology optimization of fluids in Stokes flow. *International Journal for Numerical Methods in Fluids*, 41:77–107, 2003.

Brennan, D. *The Numerical Simulation of Two-Phase Flows in Settling Tanks*. PhD thesis, Imperial College London, 2001.

Burt, D. J. *Improved Design of Settling Tanks using an Extended Drift Flux model*. PhD thesis, University of Bristol, 2010.

Burt, D. J., Corton, M., Hetherington, D., and Balmforth, D. Multiphase Modelling and the Prediction of Retention Efficiency in a Side Weir CSO. In *ASCE 9th International Conference on Urban Drainage*, Oregon, USA, 2002.

CFD Direct. OpenFOAM User Guide, 2017. URL `https://cfd.direct/openfoam/user-guide/`.

CFD Online. Turbulence free-stream boundary conditions, 2014. URL `https://www.cfd-online.com/Wiki/Turbulence_free-stream_boundary_conditions`.

City of Akron. Water Reclamation Facility, 2016. URL `http://www.akronohio.gov/cms/sewer/wrf_treatprocess/index.html`.

Clarke, D. A. A Primer on Tensor Calculus. Seminar, 2011. URL http://ap.smu.ca/~dclarke/home/documents/byDAC/tprimer.pdf.

Clift, R., Grace, J. R., and Weber, M. E. *Bubbles, Drops and Particles*. Academic Press, London, 1978.

Coderre, A. Simulation du comportement d'unités de décantation primaire: évaluation du modèle de Takacs et al. (1991). Master's thesis, Université Laval, Quebec, Canada, 1999.

Crabtree, R. W. Sediments in Sewers. *Water and Environment Journal*, 3(6): 569–578, 1989.

CST Wastewater Solutions. Lamella Plate System, 2014. URL http://www.cstwastewater.com/lamella-plate-system/.

Dahl, C. *Numerical modelling of flow and settling in secondary settling tanks*. PhD thesis, Aalborg University, Denmark, 1993.

Dudley, J. An Evaluation of a Pilot Lamella Settler and the Swirl-Flo™ Separator. Technical Report UC 2356, Water Research Centre, Swindon, UK, 1994.

Egarr, D. A. *Studies of Fluidic Systems for Environmental Applications*. PhD thesis, University of Wales, 2005.

Egarr, D. A., Faram, M. G., O'Doherty, T., Phipps, D. A., and Syred, N. Computational fluid dynamic prediction of the residence time of a vortex separator applied to disinfection. In *4th International Conference on Sewer Processes & Networks*, pages 49–56, Madeira, Portugal, 2004.

Egarr, D. A., Faram, M. G., Guymer, I., O'Doherty, T., and Syred, N. Use of computational fluid dynamics to assess the disinfection performance of a combined sewer overflow treatment chamber. In *10th International Conference on Urban Drainage*, Copenhagen, Denmark, 2005a.

Egarr, D. A., Faram, M. G., O'Doherty, T., Phipps, D. A., and Syred, N. Computational fluid dynamic prediction of the residence time of a vortex separator applied to disinfection. *Water Science & Technology*, 52(3):29–36, 2005b.

Egarr, D. A., Faram, M. G., O'Doherty, T., Phipps, D. A., and Syred, N. Computational fluid dynamic prediction of the residence time distribution of a prototype hydrodynamic vortex separator operating with a base flow component. *Proceedings of the Institution of Mechanical Engineers, Part E: Journal of Process Mechanical Engineering*, 219(1):53–67, 2005c.

Egarr, D. A., G., F. M., O'Doherty, T., and Syred, N. Experimental study of a hydrodynamic vortex separator. *Process Mechanical Engineering*, 223(1):1–10, 2009. ISSN 0954-4089. doi: 10.1243/09544089JPME225. URL http://orca.cf.ac.uk/7704/.

EPA. Wastewater Technology Fact Sheet: Screening and Grit Removal, 2003. URL https://www3.epa.gov/npdes/pubs/final_sgrit_removal.pdf.

Ergun, S. Flow through Packed Columns. *Chemical Engineering Progress*, 48 (2):89–94, 1952.

Faram, M. G. and Andoh, R. Y. G. Application of simulation and predictive techniques for the evaluation of hydrodynamic separators. In *Wastewater Treatment: Standards and Technologies to Meet the Challenges of the 21st Century, CIWEM/AETT Millennium Conference, Wakefield, UK*, pages 223–230, Leeds, UK, 2000.

Faram, M. G. and Harwood, R. CFD for the water industry; The role of CFD as a tool for the development of wastewater treatment systems. In *Fluent Users' Seminar*, Sheffield, UK, 2000.

Faram, M. G. and Harwood, R. Assessment of the effectiveness of stormwater treatment chambers using computational fluid dynamics. In *9th International Conference on Urban Drainage*, Portland, Oregon, USA, 2002.

Faram, M. G. and Harwood, R. A method for the numerical assessment of sediment interceptors. *Water Science & Technology*, 47(4):167–174, 2003.

Faram, M. G., James, M. D., and Williams, C. A. Wastewater Treatment Using Hydrodynamic Vortex Separators. In *CIWEM/AETT 2nd National Conference, Wakefield, UK*, pages 79–87, 2004.

Farrell, P. E. Differentiating functionals, 2016. URL http://www.dolfin-adjoint.org/en/latest/documentation/maths/3-gradients.html.

Funke, S. W., Farrell, P. E., and Piggott, M. D. Tidal turbine array optimisation using the adjoint approach. *Renewable Energy*, 63:658–673, 2014.

Gernaey, K., Vanrolleghem, P. A., and Lessard, P. Modeling of a reactive primary clarifier. *Water Science & Technology*, 43(7):73–81, 2001.

Giannakoglou, K. C. Adjoint-based Optimization – Methods & Applications. Seminar, 2012. URL http://www.dansis.dk/Filarkiv/pdf-filer/2012/3/Giannakoglou_Adj-Dansis12.pdf.

Gidaspow, D. *Multiphase flow and fluidization, Continuum and kinetic theory descriptions*. Academic Press, Boston, 1994.

Giles, M. B. and Pierce, N. A. An Introduction to the Adjoint Approach to Design. *Flow, Turbulence and Combustion*, 65:393–415, 2000.

Gronowska-Szneler, M. A. *Dimensioning of Vortex Separators*. PhD thesis, Gdańsk University of Technology, 2015.

Grossberg, S., Jarman, D. S., and Tabor, G. R. Validation Study of a Hydrodynamic Vortex Separator. In *Workshop on Advances in Numerical Modelling of Hydrodynamics*, Sheffield University, UK, 2015.

Guest, J. K. and Prévost, J. H. Topology optimization of creeping flows using a Darcy-Stokes finite element. *International Journal for Numerical Methods in Engineering*, 66(3):461–484, 2006.

Harlow, F. H. Fluid dynamics in Group T-3 Los Alamos National Laboratory: (LA-UR-03-3852). *Journal of Computational Physics*, 195(2):414–433, 2004.

Herrick, P., Neumayer, A., and Osei, K. Grit Particle Settling – Refining The Approach. Guest Column, 2015. URL `https://www.wateronline.com/doc/grit-particle-settling-refining-the-approach-0001`.

Hess, J. L. and Smith, A. M. O. Calculation of Potential Flow About Arbitrary Bodies. *Progress in Aerospace Sciences*, 8:1–138, 1967.

Hoffmann, K. A. and Chiang, S. T. *Computational Fluid Dynamics, Volume 3*. Engineering Education Systems, Fourth edition, 2000.

Holst, T. L. and Pulliam, T. H. Transonic Wing Shape Optimization using a Genetic Algorithm. In *IUTAM Symposium Transsonicum IV*, pages 245–252, Göttingen, Germany, 2003. Kluwer Academic Publishers.

Ishii, M. and Zuber, N. Drag Coefficient and Relative Velocity in Bubbly, Droplet and Particulate flow. *J. AIChE.*, 25:843–854, 1979.

Jameson, A. Aerodynamic design via control theory. *Journal of Scientific Computing*, 3(3):233–260, 1988.

Jameson, A. and Caughey, D. A Finite Volume Method for Transonic Potential Flow Calculations. In *Third AIAA Computational Fluid Dynamics Conference*, pages 77–635, Alburquerque, New Mexico, USA, 1977.

Jarman, D. S. *A Study of the Design of Cylindrical Vortex Flow Controls for Use in Urban Drainage Systems*. PhD thesis, University of Exeter, 2011.

Jarman, D. S., Faram, M. G., Tabor, G., and Butler, D. A review of the opportunities presented through the application of computational fluid dynamics (CFD) to water management challenges. In *Water Management Challenges in Global*

Change: CCWI2007 and SUWM2007 conference, pages 151–157, De Montfort University, Leicester, UK, 2007.

Jarman, D. S., Faram, M. G., Butler, D., Tabor, G., Stovin, V. R., Burt, D., and Throp, E. Computational fluid dynamics as a tool for urban drainage system analysis: A review of applications and best practice. In *11th International Conference on Urban Drainage*, Edinburgh, UK, 2008.

Jasak, H. *Error Analysis and Estimation for the Finite Volume Method with Applications to Fluid Flows*. PhD thesis, Imperial College London, 1996.

Jones, W. P. and Launder, B. E. The prediction of laminarization with a two-equation model of turbulence. *International Journal of Heat and Mass Transfer*, 15(3):301–314, 1972.

LAHC. Accuracy and Precision, 2017.

Launder, B. E. and Sandham, N. D. *Closure Strategies for Turbulent and Transitional Flows*. Cambridge University Press, 2002.

Launder, B. E., Reece, G. J., and Rodi, W. Progress in the development of a reynolds-stress turbulence closure. *Journal of Fluid Mechanics*, 68(3):537–566, 1975.

Li, Y. *Development of Design Basis for Hydrodynamic Vortex Separators*. PhD thesis, Rutgers, The State University of New Jersey, 2009.

Lions, J. L. *Optimal Control of Systems Governed by Partial Differential Equations*. Springer, 1971.

Lyczkowski, R. W. The History of Multiphase Computational Fluid Dynamics. *Industrial & Engineering Chemical Research*, 49(11):5029–5036, 2010.

Menter, F. and Esch, T. Elements of industrial heat transfer prediction. In *16th Brazilian Congress of Mechanical Engineering (COBEM)*, 2001.

Metcalf and Eddy. *Wastewater Engineering: Treatment and Reuse*. McGraw-Hill, Fourth edition, 2003.

Moriasi, D. N., Arnold, J. G., Van Liew, M. W., Bingner, R. L., Harmel, R. D., and Veith, T. L. Model Evaluation Guidelines for Systematic Quantification of Accuracy in Watershed Simulations. *American Society of Agricultural and Biological Engineers*, 50(3):885–900, 2007.

Nilsson, U. Description of adjointShapeOptimizationFoam and how to implement new objective functions. Master's thesis, Chalmers University of Technology, Gothenburg, Sweden, 2014.

O'Doherty, T., Egarr, D. A., Faram, M. G., Guymer, I., and Syred, N. Residence time study of a hydrodynamic vortex separator applied to predicting disinfection performance. *Proceedings of the Institution of Mechanical Engineers, Part E: Journal of Process Mechanical Engineering*, 223(3):113–122, 2009a.

O'Doherty, T., Egarr, D. A., Faram, M. G., Guymer, I., and Syred, N. Assessment of residence time in a hydrodynamic vortex separator by applying distribution models. *Proceedings of the Institution of Mechanical Engineers, Part E: Journal of Process Mechanical Engineering*, 223(3):179–188, 2009b.

OpenSource. What is open source?, 2017. URL `https://opensource.com/resources/what-open-source`.

Othmer, C. A continuous adjoint formulation for the computation of topological and surface sensitivities of ducted flows. *International Journal for Numerical Methods in Fluids*, 58(8):861–877, 2008.

Othmer, C. Adjoint methods for car aerodynamics. *Journal of Mathematics in Industry*, 4(6), 2014.

Othmer, C., Kaminski, T., and Giering, R. Computation of topological sensitivities in fluid dynamics: Cost function versatility. In *ECCOMAS CFD 2006*, TU Delft, The Netherlands, 2006.

Othmer, C., de Villiers, E., and Weller, H. G. Implementation of a continuous adjoint for topology optimization of ducted flows. *AIAA-2007-3947*, 2007.

Pathapati, S.-S. and Sansalone, J. J. CFD Modeling of a Storm-Water Hydrodynamic Separator. *Journal of Environmental Engineering*, 135(4):191–202, 2009.

Pironneau, O. On optimum design in fluid mechanics. *Journal of Fluid Mechanics*, 64(1):97–110, 1974.

Polak, E. *Optimization : Algorithms and Consistent Approximations*. Springer-Verlag, 1997.

Pugh, E. M. and Winslow, G. H. *The Analysis of Physical Measurements*. Addison-Wesley, 1966.

Rozvany, G. I. N. *Topology Optimization in Structural Mechanics*. Springer-Verlag, 1997.

Salim, M. S. and Cheah, S. C. Wall y^+ Strategy for Dealing with Wall-bounded Turbulent Flows. In *Proceedings of the International MultiConference of Engineers and Computer Scientists Vol II*, pages 978–988, Hong Kong, 2009.

Sansalone, J. J. and Pathapati, S.-S. Particle dynamics in a hydrodynamic separator subject to transient rainfall-runoff. *Water Resources Research*, 45, 2009.

Schmitt, V., Dufresne, M., Vazquez, J., Fischer, M., and Morin, A. Separation efficiency of a hydrodynamic separator using a 3D computational fluid dynamics multiscale approach. *Water Science & Technology*, 69(5):1067–1073, 2014.

Shih, T. H., Liou, W. W., Shabbir, A., Yang, Z., and Zhu, J. A new $k - \epsilon$ eddy-viscosity model for high reynolds number turbulent flows. *Computers and Fluids*, 24(3):227–238, 1995.

Smisson, B. Design, Construction and Performance of Vortex Overflows. In *Institute of Civil Engineers. Symposium on Storm Sewage Overflows*, pages 99–110, London, 1967.

Stück, A. *Adjoint Navier–Stokes Methods for Hydrodynamic Shape Optimisation*. Schriftenreihe Schiffbau, 2012.

Stück, A. and Rung, T. Adjoint RANS with filtered shape derivatives for hydrodynamic optimisation. *Computers and Fluids*, 47:22–32, 2011.

Symon, K. *Mechanics*. Addison-Wesley, Third edition, 1971.

Takacs, I., Patry, G. G., and Nolasco, D. A dynamic model of the clarification-thickening process. *Water Research*, 25(10):1263–1271, 1991.

Tukovic, Z. personal communication, 2015.

Ubbink, O. *Numerical prediction of two fluid systems with sharp interfaces*. PhD thesis, Imperial College London, 1997.

UK Technical Advisory Group. Proposals for Further Environmental Quality Standards for Specific Pollutants, 2012. URL `https://www.wfduk.org/sites/default/files/Media/UKTAG%20Specific%20Pollutants_final_011412.pdf`.

Veerapen, J. P., Lowry, B. J., and Couturier, M. F. Design methodology for the swirl separator. *Aquacultural Engineering*, 2005.

Versteeg, H. K. and Malalasekera, W. *An Introduction to Computational Fluid Dynamics: The Finite Volume Method*. Pearson Education Limited, 2007.

Vesilind, A. P. Discussion of ”Evaluation of activated sludge thickening theories”, by R. I. Dick and B. B. Ewing. *Journal of the Sanitary Engineering Division*, 94: 185–191, 1968.

WEF. *Clarifier Design*. McGraw-Hill, Alexandria, Virginia, Second edition, 2005.

Wilcox, D. C. *Turbulence Modeling for CFD*. D C W Industries, Third edition, 2006.

Witt, P. J. and Perry, J. H. A Study in Multiphase Modelling of Fluidised Beds. In *Computational Techniques and Applications*, Swinburne, Australia, 1995.

Yakhot, V., Orszag, S. A., Thangam, S., Gatski, T. B., and Speziale, C. G. Development of turbulence models for shear flows by a double expansion technique. *Physics of Fluids A*, 4(7):1510–1520, 1992.

Ying, G., Sansalone, J. J., Pathapati, S.-S., Garofalo, G., Maglionico, M., Bolognesi, A., and Artina, A. Stormwater treatment: examples of computational fluid dynamics modeling. *Frontiers of Environmental Science & Engineering*, 6(5): 638–648, 2012.

Zingg, D. W., Nemec, M., and Pulliam, T. H. A comparative evaluation of genetic and gradient-based algorithms applied to aerodynamic optimization. *European Journal of Computational Mechanics*, 17(1–2):103–126, 2008.